# ENGINEERING APPLICATIONS OF INDEPENDENT COMPONENT ANALYSIS

*A Thesis Submitted*
*in Partial Fulfillment of the Requirements*
*for the Degree of*

**Master of Technology**

*by*

SANDEEP KUMAR YADAV

*to the*

NUCLEAR ENGINEERING AND TECHNOLOGY PROGRAMME

**INDIAN INSTITUTE OF TECHNOLOGY,**
**KANPUR**

DECEMBER, 2004

# Contents

# ABSTRACT

The fundamental area of research in this thesis is Independent Component Analysis (ICA). ICA is a tool for discovering structure and patterns in data by factoring a multidimensional data distribution into a product of one-dimensional, statistically independent component distributions. Statistical independence is equivalent to information-theoretic independence. Therefore, if the original $M$ dimensional data distribution is factored into $L \leq M$ independent components, then the $M$ streams of observed numbers, which may have structure and information encoded across them, are transformed into $L$ independent streams of numbers which will have structure and information encoded only within each stream. This has the effect of making patterns within the original data more cogent.

Traditional ICA methods, however, can be limited in the flexibility of their decompositions, particularly in the modeling of component distributions and ascertaining the most appropriate number of components to adequately represent the observed data distribution. This thesis aims to develop a more flexible formulation of ICA which can overcome these limitations, allowing ICA to be applied to a wider range of data analysis problems. In this thesis we have solved the limitations of the all the existing ICA models by suggesting a new approach called higher order- independent component analysis (HO-ICA) and solved the different problems of real world data.

# Chapter 1

# INTRODUCTION

**"Independent component analysis is a recent and powerful addition to the methods that scientists and engineers have available to explore large data sets in high dimensional space."**

Recently, blind source separation by Independent Component Analysis (ICA) has received attention because of its potential applications in signal processing such as in speech recognition systems, telecommunications and medical signal processing. The goal of ICA is to recover independent sources given only sensor observations that are unknown linear mixtures of the unobserved independent source signals. In contrast to correlation-based transformations such as Principal Component Analysis (PCA), ICA not only decorrelates the signals (2nd-order statistics) but also reduces higher-order statistical dependencies, attempting to make the signals as independent as possible. In other words, "ICA is a way of finding a linear non-orthogonal co-ordinate system in any multivariate data. The directions of the axes of this co-ordinate system are determined by both the second and higher order statistics of the original data. The goal is to perform a linear transform which makes the resulting variables as statistically independent from each other as possible."

The technique of ICA is a relatively new invention. It was for the first time introduced in early 1980's in the context of artificial neural networks. In mid-1990's, some highly successful new algorithms were introduced by several research groups, together with impressive demonstrations on problems like the cocktail-party effect, where the individual speech waveforms are found from their mixtures. ICA became one of the exciting new topics both in the field of neural networks, especially unsupervised learning, and more generally in advanced statistics and signal processing.

Two different research communities have considered the analysis of independent components. On one hand, the study of separating mixed sources observed in an array of sensors has been a classical and difficult signal procossing problem. The seminal work on blind source separation was by Herault and Jutten (1986) where they introduced an adaptive algorithm in a simple feedback architecture that was able to separate several unknown independent sources. Their approach has been further developed by Jutten and Herault (1991), Karhunen and Joutsensalo (1994), Cichocki, Unbehauen and Rummert (1994). Comon (1994) elaborated the concept of independent component analysis and proposed cost functions related to the approximate minimization of mutual information between the sensors.

In parallel to blind source separation studies, unsupervised learning rules based on information-theory were proposed by Linsker (1992). The goal was to maximize the mutual information between the inputs and outputs of a neural network. This approach is related to the principle of redundancy reduction suggested by Barlow (1961) as a coding strategy in neurons. Each neuron should encode features that are as statistically independent as possible from other neurons over a natural ensemble of inputs; decorrelation as a strategy for visual processing was explored by Atick (1992). Nadal and Parga (1994) showed that in the low-noise case, the maximum of the mutual information between the input and output of a neural network implied that the output distribution was factorial; that is, the multivariate probability density function (p.d.f.) can be factorized as a product of marginal p.d.f.s. Roth and Baram (1996) and Bell and Sejnowski (1995) independently derived stochastic gradient learning rules for this maximization and applied them, respectively, to forecasting, time series analysis, and the blind separation of sources. Bell and Sejnowski (1995) put the blind source separation problem into an information-theoretic framework and demonstrated the separation and deconvolution of mixed sources. Their adaptive methods are more plausible from a neural processing perspective than the cumulant-based cost functions proposed by Comon (1994). A similar adaptive method for source separation was proposed by Cardoso and Laheld (1996).

Other algorithms for performing ICA have been proposed from different viewpoints. Maximum Likelihood Estimation (MLE) approaches to ICA were first proposed by Gaeta and Lacoume (1990) and elaborated by Pham (1992). Perlmutter and Parra (1996), MacKay (1996) and Cardoso (1997) showed that the infomax approach of Bell and Sejnowski (1995) and the maximum likelihood estimation approach are equivalent. Girolami and Fyfe (1997b,c), motivated by information-theoretic indices for Exploratory Projection Pursuit (EPP) used marginal negentropy {A general term for negentropy is relative entropy (Cover and Thomas, 1991)} as a projection index and showed that kurtosis-seeking projection pursuit will extract one of the underlying sources from a linear mixture. A multiple output EPP network was developed to allow full separation of all the underlying sources (Girolami and Fyfe, 1997c). Nonlinear PCA algorithms for ICA which have been developed by Karhunen and Joutsensalo (1994), Xu (1993) and Oja (1997) can also be viewed from the infomax principle since they approximately minimize the sum of squares of the fourth-order marginal cumulants (Comon, 1994) and therefore approximately minimize the mutual information of the network outputs (Girolami and Fyfe, 1997a). Bell and Sejnowski (1995) have pointed out a similarity between their infomax algorithm and the Bussgang algorithm in signal processing and Lambert (1996) elucidated the connection between three different Bussgang cost functions. Lee et al. (1998) show how the Bussgang property relates to the infomax principle and how all of these seemingly different approaches can be put into a unifying framework for the source separation problem based on an information theoretic approach.

The original infomax learning rule for blind separation by Bell and Sejnowski (1995) was suitable for super-Gaussian sources. Girolami and Fyfe (1997b) derive, by choosing negentropy as a projection pursuit index, a learning rule that is able to blindly separate mixed sub- and super-Gaussian source distributions. Lee, Girolami and Sejnowski (1997) show that the learning rule is an extension of the infomax principle satisfying a general stability criterion and preserving the simple architecture of Bell and Sejnowski (1995). When optimized using the natural gradient (Amari, 1997), or equivalently the relative gradient (Cardoso and Laheld, 1996), the learning rule gives superior convergence. Simulations and results on real-world physiological data show the power of the proposed methods (Lee, Girolami and Sejnowski, 1997).

Extensive simulations have been performed to demonstrate the power of the learning algorithm. However, instantaneous mixing and unmixing simulations are problems and the challenge lies in dealing with real world data. Makeig et al. (1996) applied the original infomax algorithm to EEG and ERP data showing that the algorithm can extract EEG activations and isolate artifacts. Jung et al. (1997) show that the extended infomax algorithm is able to linearly decompose EEG artifacts such as line noise, eye blinks, and cardiac noise into independent components with sub- and super-Gaussian distributions. McKeown et. al. (1997) have used the extended ICA algorithm to investigate task-related human brain activity in fMRI data. By determining the brain regions that contained significant amounts of specific temporally independent components, they were able to specify the spatial distribution of transiently task-related brain activations. Other potential applications may result from exploring independent features in natural images. Bell and Sejnowski (1997) suggest that independent components of natural scenes are edge filters. The filters are localized, mostly oriented and similar to Gabor like filters. The outputs of the ICA filters are sparsely distributed. Bartlett and Sejnowski (1997) and Gray, Movellan and Sejnowski (1997) demonstrate the successful use of the ICA filters as features in face recognition tasks and lippreading tasks respectively.

For these applications, the instantaneous mixing model may be appropriate because the propagation delays are negligible. However, in real environments substantial time-delays may occur and an architecture and algorithm is needed to account for the mixing of time-delayed sources and convolved sources. The multichannel blind source separation problem has been addressed by Yellin and Weinstein (1994) and Ngyuen and Jutten (1995) and others based on 4-th order cumulants criteria. An extension to time-delays and convolved sources from the infomax viewpoint using feedback architecture has been developed by Torkkola (1996). Lee, Bell and Lambert (1997) have extended the blind source separation problem to a full feedback system and a full feed forward system. The feed forward architecture allows the inversion of non-minimum phase systems. In addition, the rules are extended using polynomial filter matrix algebra in the frequency domain (Lambert, 1996). The proposed method can successfully separate voices and music recorded in a real environment. Lee, Bell and Orglmeister (1997) show that the recognition rate of an automatic speech recognition system is increased after separating the speech signals. L. B. Almeida (2003) proposed the MISEP an ICA technique for linear and non-linear mixtures, which is based on the minimization of mutual information of the estimated components.

Since ICA is restricted and relies on several assumptions researchers have started to tackle a few limitations of ICA. One obvious but non-trivial extension is the nonlinear mixing model. In (Hermann and Yang, 1996; Lin and Cowan, 1997; Pajunen, 1997) nonlinear components are extracted using self-organizing-feature-maps (SOFM). Other researchers (Burel, 1992; Lee, Koehler and Orglmeister, 1997; Taleb and Jutten, 1997; Yang, Amari and Cichocki, 1997) have used a more direct extension to the previously presented ICA models. They include certain flexible nonlinearities in the mixing model and the goal is to invert the linear mixing matrix as well as the nonlinear mixing. Hochreiter and Schmidhuber (1998) have proposed low complexity coding and decoding approaches for nonlinear ICA. More recently, MISEP is a generalization of popular INFOMAX technique. Francis R. Bach and Michael I. Jordan (2003) proposed tree-dependent component analysis (TCA) provides a tractable and flexible approach to weakening the assumption of independence in ICA.

Other limitations such as the under-determined problem in ICA, i.e. having fewer sensors than sources and noise models in the ICA formulation are subject to current research efforts.

We have proposed a low complexity coding, flexible ICA algorithm and given the name as higher order – independent component analysis (HO-ICA). By applying this algorithm we have solved the nonlinearities, over-determined and under-determined problems.

In the field of image processing we have introduced some of the new ideas like number plate and hidden face detection, image compression etc. by applying proposed algorithm.

ICA is a fairly new and a generally applicable method to several challenges in signal processing. It reveals a diversity of theoretical questions and opens a variety of potential applications. Successful results in EEG, fMRI, speech recognition and face recognition systems indicate the power and optimistic hope in the new paradigm.

## 1.1 Literature review

The way data is presented very much influence what pattern can be seen and how much information can be extracted from it. A primary goal in pattern recognition is to find some intrinsic coordinate system in which the data structure is most apparent. Traditionally, second –order information in the guise of the data covariance structure has been used to construct these coordinate frames, yielding Gaussian- based techniques such as Principal Component Analysis (PCA) and Factor Analysis. With the increase of computational power over recent years, however, the use of higher- order information has become feasible leading to more sophisticated non- Gaussian methods such as Exploratory Projection Pursuit and Independent Component Analysis. PCA [1] is widely used statistical tool for representing and compressing data by finding an intrinsic orthogonal coordinates system for the observation data that preserves the maximum amount of variance possible. PCA only works up to the second order statistics, so factoring non-Gaussian densities is not possible with second order information. Another method that uses higher- order information for data representation and one intimately linked with ICA is Exploratory Projection Pursuit (EPP) [2, 3, and 4]. The aim of EPP is to find a low-dimensional (order 2 or 3) projection of the observation data that aids the visualization of structure not included in the data covariance. It [3, 4] it has been shown that Gaussian is the least interesting of all distributions. Therefore, the notion of 'non-Gaussianity' can form the basis of constructing an index. Any quantity that measures the non-Gaussianity of a projection can serve as measurement of 'interestingness'. Such measure includes functions of statistical moments, cumulants [5] and negentropy. Different projections of the data may find different structures, so the EPP Procedure is usually iterated a number of times. After each index maximization, the direction found is either extracted or 'Gaussianated' [2] and the EPP process repeated to find further directions. This continues until no more directions remain. The set of projections found form a collection of directions or components each containing separate structural information. It can be shown [6] that, in the low observation-noise limit, these components are in fact the independent components of the distribution. Indeed, ICA algorithms can be derived by using non-Gaussianity as a measure of independence [7, 8, and 9]. Although a far more sophisticated tool than second-order methods such as PCA, EPP can be limited in application. Measuring non-Gaussianity utilizing the above quantities directly is computationally expensive so approximations are usually made. These approximations have the effect of smoothing the data distribution, so subtle structure or structure close to Gaussianity is often lost. There is no explicit way of modeling observation noise, so noisy data can cause spurious directions to be found. EPP algorithms based on cumulants are also sensitive to distribution outliers so can be less efficient with noisy data. The effectiveness of EPP decreases with increasing projection dimensionality due to the computational cost of computing the pursuit indices and, in particular, performing the structure removal step [2]. As discussed above, the set of 'interesting ≡ non-Gaussian' components is equivalent to the set of independent components. Independence can be quantified in a variety of ways, not just by measuring non-Gaussianity. Therefore, finding projections using independence as the criterion of interest can overcome many of these problems. Furthermore, maximizing an objective functions measuring independence obviates the need for a removal step as all the independent directions can be found simultaneously. All the algorithms of ICA developed by different researchers have some limitations. In this work we have tried to overcome by these limitations by applying the new algorithm of ICA, in the coming chapters we will discuss about all these things in detail.

## 1.2 Problem definition

In Image processing , compression Schemes are aimed to reduce the transmission rate for still (or fixed images), while maintaining a good level of visual quality , There are some parameters to measure the quality of images like Peak Signal to Noise Ratio (PSNR) and Root Mean Square Error (RMSE). There are so many methods for the image compression like Wavelet Compression, JPEG Compression, and Fractal Compression etc. But all these techniques are lossy in nature.

In the case of Signal Processing (Audio and Video signals) noise can create deterrent to understand data. In case of low PSNR it becomes exceedingly difficult to identify the signal. Situation becomes further complicated in case only aggregated mixture of different sources is available without knowing their origin. Some attempts have been made to determine sources which have contributed to the mixture. These approaches include principle component analysis (PCA), projection pursuit (PP) and independent component analysis (ICA). ICA is potentially more rigorous tool compare to others because it deals with the higher order statistics.

In the area of security, face recognition, matching the face in the already stored database, finger print detection, number plate detection, fraud detection, feature extraction ,etc. are the major problems in the real world. By applying the ICA one can overcome from all these problems.

The blind source separation problem is to extract the underlying source signals from the set of linear mixtures, where the mixing matrix is unknown. This problem is common in acoustics, radio, medical signal, image processing and hyper spectral imaging, etc.

Network traffic is a major problem in the area of communication. ICA can be applied to the compression of data.

All the ICA algorithms developed by many researchers so far for solving the above stated problems were using the different algorithm of Independent Component Analysis like Fast ICA, Gradient Based, etc. All these algorithms are having some drawbacks like order of inputs, non-linearity and complexity. We have developed the new ICA technique keeping these limitations in the mind and solved all the drawbacks.

## 1.3 Organization of Thesis

The objective of the thesis is Engineering Application of Independent component analysis. There are so many algorithm of ICA to solve the above mentioned problems. In this thesis work I have tried to solve the below mentioned problems.

1. Blind Source Separation
2. Number Plate Detection
3. Hidden face detection
4. Image Feature Extraction
5. Image Compression

The second chapter discusses about the need of ICA, what is ICA, different algorithms including one we are currently pursuing and applications of ICA. Third chapter contains the application of proposed algorithm for the solution of the real world problems mentioned as above. Finally the fourth chapter contains the conclusion and future scope.

# CHAPTER-2

# NEED OF ICA

## 2.1 Introduction

Science is all about understanding what is going on around us. Applied Science, like engineering, is especially focused on how this understanding can be exploited to create systems that are useful to us. ICA is treated as a way to enhance our knowledge about measured signals. It assumes that many signals that can be measured actually originate from independent sources and provides us with a method to retrieve these sources. In ICA, these independent sources are called as independent components (ICs). Knowing the independent components and knowing in which way they contribute to what has been measured can give insight to the process that generated the data. If nothing is known about the process, it may be possible to retrieve some of the model parameters.

A finite number of independent components can be linearly combined in infinitely many ways. Natural images all have at their source the same independent Components. The way each IC contributes to the total signal then determines the image. In this way, it would be possible to compress natural images, based on their independent components.

The most well known application of ICA is the Blind Source Separation (BSS) problem.

A famous application of BSS is the 'cocktail party problem'. Suppose that we find ourselves hanging on a cocktail party with many other people hanging around. If every person holding a microphone to record his voice, we would find out that each microphone did not only record the voice of its owner, but also the voices of all other speakers at the cocktail party. So each microphone records a linear mixture of all the speech- signals in the room added some noise. ICA is able to determine these independent components (the individual speech- signals) based on only the information from the micro phones.

From the above stated examples, it can be noticed that ICA is about to extracting features appropriate for a given application. Intuitively, we may also sense that the ICs of a certain dataset can be appropriate features in general. Finding the independent components of certain observations may be very useful for several applications.

## 2.2 **Principal Component Analysis**

Principal Component Analysis is a method for reducing the dimensionality of data, that is, representing a set of m dimensional vectors with n ‹ m components for each vector so that information is lost " as minimum as possible". In terms of linear algebra the problem of dimension reduction consists of finding a new basis for the data so that if we drop out or zero some of the components in the new basis, the reconstruction error is as small as possible. In practice the most easily computable norm for establishing an optimality criterion for the conservation of the information is the mean square norm. If random vector $x$ represents the original data and the estimate in the new basis is $\hat{x}$, the reconstruction error random vector $e = x - \hat{x}$. Then the goal is to minimize $\mathrm{E}\{\|e\|^2\}$. Assume that $\mathrm{E}\{x\} = 0$.it is proved in standard references [10, 11] that the optimal new basis consists of the eigenvectors of the covariance matrix of $x$

$$\mathrm{cov}\{x\} = \mathrm{E}\{xx^{\mathrm{T}}\}.$$

And that the eigenspaces that retain the most significant amount of information are those that correspond to the largest eigenvalues.These eigenvalues also represent the variances of projections onto the subspace, and the PCA directions are optima of variances of projections. We also know that the PCA basis is orthogonal, because for an invertible real symmetric matrix – like $\mathrm{cov}\{x\}$ in the nondegenerate case -- there exists a set of orthogonal eigenvectors.

Let us denote the set of eigenvectors/eigenvalues pairs of $\mathrm{cov}\{x\}$ by $\{(e_1, \xi_1), ..., (e_n, \xi_n)\}$ and let $E = [e_1 ... e_n]$ and $D = \mathrm{diag}(\xi_1, ........., \xi_n)$ (a matrix with elements $\xi_1, ........., \xi_n$ on diagonal). If the data is projected onto the new basis, the new components are uncorrelated. This can be seen by considering the components in the new basis, given by $x_{PCA} = E^T x$, for which

$$\mathrm{cov}\{x_{PCA}\} = \mathrm{E}\{x_{PCA} x_{PCA}^T\} = \mathrm{E}\{E^T xx^T E\}$$

$$= E^T \mathrm{cov}\{x\} E = E^T EDE^T E = D$$

By eigenvalue decomposition of the covariance matrix, $\text{cov}\{x_{PCA}\} = EDE^T$, and $E$ is orthogonal because the eigenvectors form an orthonormal vector set.

## 2.3 The concept of independence:

Two events A and B in event space of an experiment are called independence if

$$\text{P (AB)} = \text{P (A) P (B)}.$$

Conditional probability P (B/A) is given by

$$\text{P(B/A)} = \frac{\text{P(AB)}}{\text{P(A)}} \tag{1}$$

We can see that independence implies that P (B/A) = P (B), if P (A) $\neq$ 0. Actually this condition is equivalent to independence, since the implication holds in the other direction by (1). Assuming that P (B) $\neq$ 0 a corresponding equivalence P (A/B) = P (A) ↔ P (AB) = P (A) P (B) can be derived by the similar procedure.

Equations P (B/A) = P (B) and P (A/B) = P (A) have an appealing intuitive interpretation: if one event happens, it does not give us any additional information about the probability of the other event. It can be proven that if A and B are independent, then all pairs $\{\text{A}, \overline{\text{A}}\} \times \{\text{B}, \overline{\text{B}}\}$ are independent [12], so independence holds for presence and absence of the event. When events are concerned, the presence or absences of the events are only the information we can get from an experiment. In terms of information, independence of two events could be described so that information about one event gives no additional information about the other one.

In the case of random variables independence has very similar interpretation. The definition of two random variables $x$ and $y$ is

$$\text{P}(x \in \text{A}, y \in \text{B}) = \text{P}(x \in \text{A})\text{P}(y \in \text{B}),$$

Where $\text{A}, \text{B} \subset R$. This is equivalent to [12]

$$f_{xy}(x,y) = f_x(x) f_y(y) \tag{2}$$

Assuming that the densities exist. The conditional density

$$f_y(y/x) = \frac{f_{xy}(x,y)}{f_x(x)}$$

Becomes $f_y(y/x) = f_y(y)$. In this case information about the value of one random variable gives us no information about the value of the other one.

The significance of independence is due to the fact that the independent components can often be processed separately.

## 2.4 Linear Model of independent component analysis

Let us assume that we have some phenomena which manifest itself through a set of $n$ independent random variables. We shall denote the combination of these variables with random vector $s = [s_1, s_2, ..., s_n]^T$. Components $s_1, s_2, ..., s_n$ are called sources and $s$ is called the source vector.

Now suppose that the original independent source components are observed via a linear process. Denote the observed random vector by $x_{obs}$. Since the process is assumed linear, the relation in between $s$ and $x_{obs}$ can be denoted as

$$x_{obs} = As. \tag{3}$$

Here matrix $A \in £(\Re^n, \Re^m)$ is referred to as a mixing matrix, since it mixes the independent sources. Its companion is the separating matrix $W \in £(\Re^n, \Re^m)$, given by equation

$$S = Wx_{obs}. \tag{4}$$

Here in this equation all the cases *m‹ n, m = n, m › n are* possible. These three cases differ significantly from each other.

Because of a need to identify the original sources, we might be interested in extracting the sources from the observed vector. One might also want to know the structure of the mixing matrix or separating matrix is referred to as independent component analysis or ICA.

In this work we are primarily concerned with the estimation of mixing and separation matrices. Looking at equation (3) we can see that the observed random vector $x_{obs}$ is a linear mixture of column vectors of matrix A, the coefficients of each vector being given by the corresponding independent source. These column vectors $a_i \in \Re^m$, $i = 1,.....,n$ are called ICA basis vectors. Correspondingly we can that in (4) each independent component is given by taking the inner product of the observed sample vector with a row vector of matrix $W$ or in other words, filtering the sample with one of the rows. For this reason the rows of matrix $W$ are called ICA filters.

From equation (4) if we can find a transform $W$ giving the independent components, then we can scale, negate and permute the independent sources with permutation matrix $P$ and scaling matrix $S$ to give us

$S\ P\ s = S\ P\ W x$ , and the new components remain independent, so matrix $SPW$ is also a separating matrix. This means that the independent sources can only be recovered up to a scale, sign and permutation [8]. In the rest of this work we assume that scaling is done so that each independent source has unit variance, that is,

$$\text{var}\{s_i\} = 1, \quad i = 1,...,n.$$

## 2.5 Preprocessing Steps

### 2.5.1 Removing correlations

Assume that our data has zero mean, that is, E$\{x\}$ = 0. If we can find a linear transformation giving relation (4), the independent components of s have zero mean as well. We assume that the data has this property, or that is has been centered by removing its mean. The covariance between two components $s_i$ and $s_j$ of vector s, i, j$\in \{1,..........,m\}$ is

$$\mathrm{E}\{s_i\, s_j\} = \mathrm{var}\{s_i\} = 1, \text{ if } \mathrm{i} = \mathrm{j}$$

$$\mathrm{E}\{s_i\, s_j\} = \mathrm{E}\{s_i\}\ \mathrm{E}\{s_j\} = 0 \quad \text{when } \mathrm{i} \neq \mathrm{j}.$$

So the covariance matrix of s is cov {s} = I, and components of s are uncorrelated. This means that independence implies uncorrelated ness is a necessary requirement for independence. However it is not a sufficient condition. The uniform distribution of two variables depicted in Figure 1(a) has been rotated by an orthogonal matrix in Figure 1(b). The original random variables are independent and uncorrelated. As we can see from the image, the rotated data is not independent, but it is uncorrelated, since uncorrelated ness is preserved under orthogonal transformations. Let $x$ be an uncorrelated random vector. Then for an orthogonal matrix P

$$\mathrm{cov}\{\mathrm{P}x\} = \mathrm{P}\,\mathrm{cov}\{x\}\,\mathrm{P}^T = \mathrm{I},$$

So P $x$ is also uncorrelated.

By the previous discussion one might conjecture that if we first decorrelate the components of the observed variables, we shall be closer to our final objective of independence. We can accomplish uncorrelated ness by transforming $x$ so that its covariance matrix will be diagonal. If in addition all components have unit variance

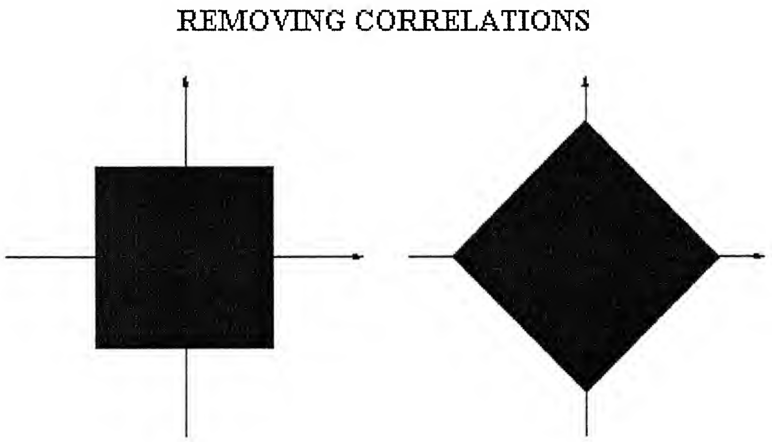


**Figure 1. An example of independence and uncorrelatedness**

(The covariance matrix is unity), a random vector is referred to as being white and the process of accomplishing this property is called whitening or sphering.

Whitening can be done using PCA basis vectors and variances along them. As discussed above let $E$ denote the matrix of principal component basis vectors of random data vector $x$, i.e., the eigenvectors of $\text{cov}\{x\}$, and $D = \text{diag}(\xi_{1,}.................,\xi_n)$ a diagonal matrix of corresponding eigenvalues. The new whitened data vector $v$ is given by

$$v = D^{-1/2}E^T x. \tag{5}$$

Matrix V = $D^{-1/2}E^T$ is a whitening matrix. The fact that $v$ is really white can be seen from

$$\text{cov}\{x\} = \text{E}\{D^{-1/2}E^T xx^T ED^{-1/2}\} = D^{-1/2}E^T \text{cov}\{x\}ED^{-1/2}$$
$$= D^{-1/2}E^T EDE^T ED^{-1/2} = I$$

PCA whitening is by no means the only possible method for whitening. If V is a whitening matrix of a random vector $x$ and P is any orthogonal matrix, then PV is a whitening matrix, because

$$\text{E}\{\text{PV}x(\text{PV}x)^T\} = \text{PE}\{\text{V}xx^T\,\text{V}^T\}\text{P}^T = \text{P}I\text{P}^T = I.$$

PCA whitening however has the advantage that optimal (in the mean square sense) dimensionality reduction can be combined with the whitening operation.

In ICA problems whitening orthogonalizes the ICA basis vectors. Suppose that n = m , so that we have as many measured components as we have independent sources, and that the data is not restricted in to a subspace of $\Re^m$.Then PCA whitening matrix $\text{V} = D^{-1/2}E^T$ exists and is invertible.

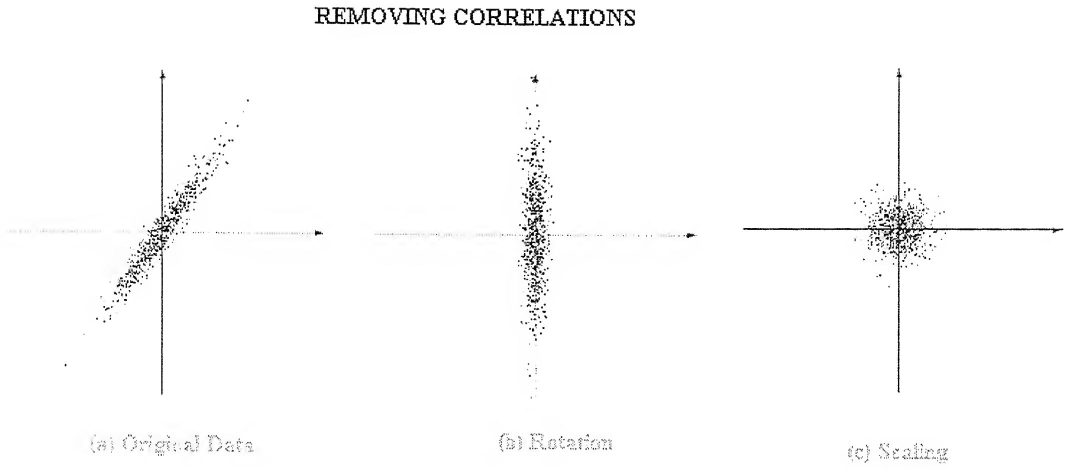


**Figure 2. Whitening**

Pre multiplying equation (3) with $V$ gives

$$v = Vx = V A\, s = B\, s. \tag{6}$$

Vector $v$ is called the whitened data vector and matrix $B \in \pounds\{\Re^n, \Re^n\}$ is the whitened mixing matrix. Noting that both $v$ and $s$ are white,

$$I = \mathrm{E}\{vv^T\} = \mathrm{E}\{Bss^T B^T\} = B\,\mathrm{E}\{ss^T\}B^T = BB^T,$$

So $B$ is orthogonal.

## 2.6 The central limit theorem and nonnormality

One way to approach the ICA problem is to try to form an optimization problem that has as its solutions the independent components. We shall call such objective functions contrast functions.

Let us approach the problem of finding contrast functions from the point of view of the central limit theorem. The central limit theorem [12] states that if we have n independent random variables $s_1,...,s_n$, the distribution of their sum $y_n = \sum_{i=1}^{n} s_i$ approaches a normal distribution as $n \to \infty$. The "speed" with which the sum approaches a Gaussian is naturally dependent of the nature of the original variables. For smooth densities even relatively small values of n may give quite normal- like densities [12]. Assume that we have two non-normal independent random variables $s_1$ and $s_2$ and $\beta \in \Re, 0 \leq \beta \leq 1$. Knowing that the sum of independent random variables approaches a normal distribution, let us make the following imprecise conjecture: with a small but nonzero $\beta$ the random variable $(1-\beta)s_1 + \beta s_2$ is "more normal" than $s_1$.

This is the intuitive idea behind contrast function measuring deviation from a normal distribution. If the previous conjecture holds, then the original independent variables are optimum points of nonnormality in all linear mixtures of the variables, because if we start to mix one variable with any of the other variables we get something closer to normal. In other words, we could in principle solve the ICA problem by finding linear combinations or projection directions that give at least locally "maximally nonnormal" random variables.

In order to quantify these ideas we introduce a measure of nonnormality called kurtosis. Typically in statistics the value of kurtosis is described to measure the peakedness of the distribution, with peaked distributions giving positive values of kurtosis and flat distribution negative values. The kurtosis is defined for a random variable $y$ as

$$k_4(y) = \mathrm{E}\{y^4\} - 3(\mathrm{E}\{y^2\})^2 \qquad (7)$$

The value of kurtosis however depends on the variance of the distribution. Variance should not affect our measure of distance from normality, since it does not affect the flatness or peakedness of the density, and there exists Gaussian distributions of all variances. So in order to use kurtosis as an optimization criterion we should ensure that the optimization is done over mixtures with a fixed variance. The variance of a projection of a random vector with mean zero is

$$\mathrm{var}\{w^T x\} = \mathrm{E}\{(\mathrm{w}^{\mathrm{T}} x)^2\} = w^T \mathrm{cov}\{x\} w.$$

This means that all vectors w should have a constant length in the norm induced by the positive definite matrix $\operatorname{cov}\{x\}$,

$$\|w\|_{\operatorname{cov}\{x\}} = w^T \operatorname{cov}\{x\} w.$$

An easy way to ensure this is fixed for all w is to have cov{x} = I (use the Euclidean norm) and $\|w\| = 1$ for all w (normalize the vectors in this norm). In other words

- X should be white (here we should note that we did not need the uncorrelated ness but rather the sphering property of whitened data) and
- We should keep $\|w\| = 1$ while optimizing the function (optimize the function over the unit sphere).

Indeed using such constraints the optimization of kurtosis usually the ICA problem. Assuming that we have the whitened situation described above, from (6) we can see that the independent components are given or $s_i = b_i^T v$, $i = 1,...,n$ where $B = [b_1,...,b_n]$. The orthogonal columns of B span $\Re^n$, so if $w \in \Re^n$, then $w = \sum_{i=1}^{n} a_i b_i$, $a_i \in \Re$, $i = 1,...,n$. If $s_1$ and $s_2$ are two independent random variables and $\beta \in \Re$, then by direct calculation the following very important properties hold for kurtosis

$$k_4(s_1 + s_2) = k_4(s_1) + k_4(s_2)$$
$$k_4(\beta s_1) = \beta^4 k_4(s_1).$$

Define $k_i = k_4(s_i)$, $i = 1,\cdots,n$, and $a = [a_1 \cdots a_n]^T$. Then

$$k_4\left(w^T v\right) = k_4(a^T B^T v) = \sum_{i=1}^{n} k_4(a_i s_i) = \sum_{i=1}^{n} a_i^{\,4} k_i. \tag{8}$$

For kurtosis to be constant function it would have to be optimized at one of the columns of B.

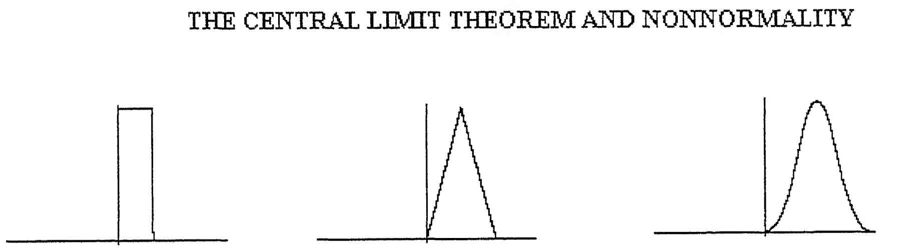
$$f(a) = \sum_{i=1}^{n} a_i^{\,4} k_i$$

Fulfills this condition, that is, the optima of $f$ are given by such unit vectors $a$ in which exactly one component differs from 0 and is 1 or -1. This proves that kurtosis is indeed a contrast.

Although kurtosis can be used to ICA problem, it has some undesirable properties. From the definition of kurtosis (7) we can see that it is very sensitive to variations in data since it is based on the fourth moment. This makes the approach sensitive to errors and outliers, because the algorithms have to estimate the statistical quantities involved. That is why we sometimes use other contrast functions to solve the ICA problem.

The following class of contrast functions is introduced in [13]. Let G be a twice differentiable nonquardratic function, $x$ a zero mean random variable and $v$ a normally distributed random variable with the same mean and variance as x. Then a contrast function based on G is

$$J_G(x) = \mathrm{E}\{G(x)\} - \mathrm{E}\{G(v)\}. \tag{9}$$

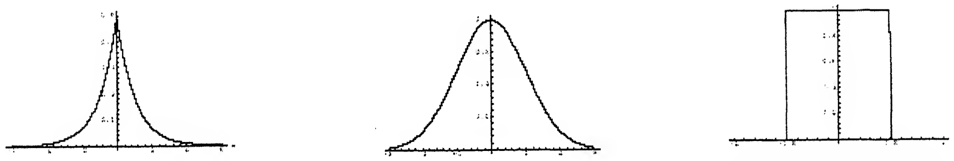


(a) A uniformaly distributed variable.

(b) Sum of two uniformly distributed variables

(c) Sum of three uniformly distributed variables

Figure 3. An illustration of central limit theorem



(c) "Peaked" distribution

(b) Gaussian distribution

(c)"Flat" distribution

Figure 4. Kurtosis as a descriptive statistic

First, G is constrained to be nonquardratic because $J_G$ would be identically zero. Second, we can immediately see that $J_G$ is when x is Gaussian. Note that if $G(t)=t^4$,(9) is reduced to the definition of kurtosis. It is shown in [13] that $J_G$ is at least locally contrast function for whitened data (the independent directions are local optima of$J_G$). One especially important choice for G is $G(t)=\ln\cosh(t)$. It is much less sensitive to errors in the data. This is illustrated in Figure 5, where the fast growth of the fourth power is contrasted with the nearly linear behavior of (lncash). It is shown in [14] that this contrast has good behavior with respect to asymptotic variance and robustness.

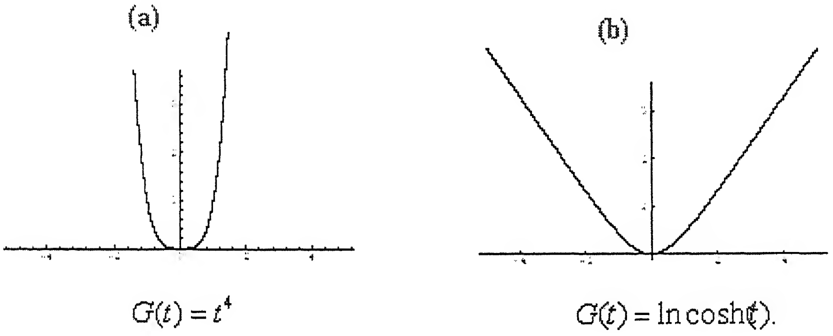


**Figure 5. Two different nonlinearities for a contrast function**

## 2.7 Cumulants and cumulant matrices

Cumulants are higher order statistics that have become increasingly popular in various signal processing tasks [15, 16].

The use of cumulants is usually contrasted with that of second - order statistics: whereas correlation and power spectrum (Fourier transform of autocorrelation) are standard tools in signal processing, cumulants and polyspectra (Fourier transform of cumulants) can be seen as corresponding statistical tools of order higher than two.

The properties of cumulants and advantage for using them are not easy to illustrate. Here we will first justify superficially the use of cumulants by giving a few of their properties even before we give their definition. Let $\text{cum}(x_{1,...},x_n)$ denote a cumulant of random variables $x_1,\cdots,x_n$.

1. Cumulants are linear in each entry (multilinear) , that is , for a random variable $y$ and scalar $\alpha$

   $$\mathrm{cum}(x_1,\cdots,x_i+y,\cdots,x_n)=\mathrm{cum}(x_1,\cdots,x_i,\cdots,x_n)+\mathrm{cum}(x_1,\cdots,y,\cdots,x_n)$$

   $$\mathrm{cum}(x_1,\cdots,\alpha x_i,\cdots,x_n)=\alpha\,\mathrm{cum}(x_1,\cdots,x_i,\cdots,x_n)$$

2. If a subset of random variables $x_1,\cdots,x_n$ is independent of the others, then $\mathrm{cum}(x_1,\cdots,x_n)=0$.
3. If $x_1,\cdots,x_n$ are Gaussian random variables and $n\geq 3$, then $\mathrm{cum}(x_1,\cdots,x_n)=0$.

The first and second property enables and eases the use of cumulants as operators. The third property makes cumulants "immune" to Gaussian noise and enables their use as measures of nonnormality. For a comprehensive presentation of these and other properties of cumulants [15, 16].

Cumulants are defined as the coefficients of the Taylor series of the second characteristic function of $x=[x_1,\cdots,x_n]^T$ at 0, that is,

$$\mathrm{cum}(x_1,\cdots,x_n)=\frac{d}{dw_1}\cdots\frac{d}{dw_n}\Psi(w_1,\cdots,w_n)\big|_{w_1=0,\cdots,w_n=0},$$

Where $\Psi\ (w_1,\cdots,w_n)=\ \ \mathrm{lnE}\{\mathrm{e}^{\sum_i w_i x_i}\}$.

The class of cumulants involving just one random variable is denoted by the letter k, so that $k_i(x)=\mathrm{cum}(x,\cdots,x)$.

## 2.8 Over- and underdetermined ICA models

Assume that matrix B has been determined by some ICA algorithm, and that we want to determine the mixing matrix A and the separating matrix W. If n = m, then we also know that V in (6) is invertible, and ICA basis vectors and filters are given by $A=V^{-1}B$ and $W=B^TV$. the situation gets more complicated when n $\neq$ m.

If the numbers of independent sources are smaller than the dimension of measured data, i.e., n ‹ m, we can use PCA to reduce our problem to a lower dimension. From (3) we note that if n ‹ m, then the measured data lies in an at most n dimensional subspace spanned by columns of A. Hence n-m PCA dimensions contain no data, and PCA can be used to constrain the data to a subspace without any loss of information.

Naturally in the real world problems data contains noise, so the data is not contained exactly in the subspace, but PCA is a good tool for such situations as well.

Let $x_{obs}$ denote the observed data vector and $D$ be the diagonal matrix of $n$ largest eigenvalues of $\mathrm{cov}\{x_{obs}\}$, and $E$ the matrix of corresponding eigenvectors. So $D \in £(\Re^n, \Re^n)$ and $E \in £(\Re^n, \Re^m)$. Then dimensionality reduction is performed with

$$x = EE^T x_{obs}\,. \tag{10}$$

When PCA whitening is used, equation (6) gives $v = D^{-\frac{1}{2}} E^T E E^T x_{obs} = D^{-\frac{1}{2}} E^T x_{obs}$, since columns of $E$ are orthogonal ($E^T E = I$).

After dimensionality reduction matrix $B$ is calculated for the reduced system. However, usually we are interested in original basis vectors of equation (3) or filters of (4), so $B$ is not our final aim. Matrix $A$ has to be solved from the equation

$$VA = B \tag{11}$$

obtained from (6). This equation can be viewed as consisting of $n$ equations

$$Va_i = b_i,\ i = 1, \cdots, n, \tag{12}$$

Where $a_i$ and $b_i$ are columns of $A$ and $B$. This is equivalent to solving

$$E^T a_i = D^{\frac{1}{2}} b_i, \qquad\qquad i = 1, \cdots, n.$$

If n ‹ m, that is, the number of sources is smaller than the number of the number of observed components, then since $E \in £(\Re^n, \Re^m)$, the equations are underdetermined and each problem has an infinite number of solutions. However, multiplication performed in operation $VA$ can be viewed as whitening of the column vectors of $A$ . Hence the solution of (11) can be viewed as whitening of the column vectors of $A$. Hence the solution of (11) can be viewed as dewhitening of column vectors of $B$. Since whitening using PCA consists of projection onto PCA subspaces and scaling with corresponding deviation, the "obvious" counterpart for dewhitening is rescaling and summation back in to $\Re^m$ , which gives us

$$A_{norm} = ED^{1/2}B \tag{13}$$

Note that $A_{norm}$ gives one solution to equation (11).Equation (13) can also be derived in another way. If $a_{part,i}$ is one solution to equation (12) then any solution $a_i$ can be given in form

$$a_i = a_{pqrt,i} + a_{null},$$

Where $a_{null} \in \mathrm{N}(V) = \{a | V_a = 0\}$. One particular choice for $a_{part,i}$ is the one with minimum Euclidean norm, given by [17].

$$a_{norm,i} = V^T(VV^T)^{-1}b_i = (D^{-1/2}E^T)^T[D^{-1/2}E^T(D^{-1/2}E^T)]^{-1}b_i$$

$$= ED^{1/2}b_i$$

This gives us (13). Solution $a_{norm,i}$ also has the property that is orthogonal to all vectors in N (V), which might be described so, that it does not contain any information form this uncertain part of the solution.

ICA basis vectors are closely related to ICA filters. After dewhitening we have from (6)

$$s = B^{-1}Vx = B^TVx,$$

Since B is orthogonal. Using (5) we get

$$W = B^TD^{-1/2}E^T. \tag{14}$$

On the other hand, from (13) we can see that

$$A_{norm}^{T} = B^{T} D^{1/2} E^{T}.$$

It should be noted that ICA filters of (14) automatically perform dimensionality reduction when applied to the original data vector $x_{obs}$, because

$$Wx_{obs} = B^{T} D^{-1/2} E^{T} x_{obs} = B^{T} D^{-1/2} E^{T} E E^{T} x_{obs}$$

$$= B^{T} D^{-1/2} E^{T} x = Wx.$$

When the number of sources are greater than the number of measured components (n › m), matrix B can not in general be solved, but can only be determined in special case [18]. Assuming that we have found B, the methods for determining A are as above, but the determination of W is more complicated now, From (6) we get $Vx = BWx$, so $W$ can be calculated by solving $BW = V$, but in this case B is not orthogonal or even square. This set of equations has again an infinite number of solutions, and like above the solution with minimum norm is chosen, giving

$$W = B^{T}(BB^{T})V.$$

To summarize the results, in the case n = m, both A and W may be resolved accurately. In case n › m, A may be solved accurately, but W contains uncertainty, because it has to be chosen from a set of possible solutions. The situation is reversed when n ‹ m. The degree of freedom for the choice of the uncertain solution in both cases is $\lfloor n - m \rfloor$

## 2.9 Negentropy:

A second very important measure of non Gaussianity is given by negentropy. Negentropy is based on the information – theoretic quantity of (differential) entropy. Entropy is the basic concept of information theory. The entropy of a random variable can be interpreted as the degree of information that the observation of the variable gives. The more “random”, i.e. unpredictable and unstructured the variable is, the larger its entropy. More rigorously, entropy is closely related to the coding length of the random variable, in fact, under some simplifying assumptions, entropy is the coding length of the random variable. For further details see [19, 12].

Entropy $H$ is defined for a discrete random variable $Y$ as

$$H(Y) = -\sum_i P(Y = a_i)\log P(Y = a_i)$$

Where the $a_i$ are the possible values of $Y$. This very well known definition can be generalized for continuous –valued random variables and vectors, in which case it is often called differential entropy. The differential entropy $H$ of a random vector $y$ with density $f(y)$ is defined as [19, 12].

$$H(y) = -\int f(y)\log f(y)dy.$$

A fundamental result of information theory is that a Gaussian variable has the largest entropy among all random variables of equal variance. This means that entropy could be used as a measure of nongaussianity.

To obtain the measure of nongaussianity that is zero for a Gaussian variable and always nonnegative, one often uses a slightly modified version of the definition of differential entropy, called negentropy. Negentropy J is defined as follows

$$J(y) = H(y_{gauss}) - H(y)$$

Where $y_{gauss}$ is a Gaussian random variable of the same covariance matrix as $y$. Negentropy is always

Non-negative and it is zero if and only if y has a Gaussian distribution.

### 2.9.1 Approximation of negentropy:

The estimation of negentropy is difficult, and therefore this contrast function remains mainly a theoretical one. In practice, some approximation has to be used.

The classical method of approximating negentropy is using higher-order moments, for example as follows [4].

$$J(y) \approx \frac{1}{12} E\{y^3\}^2 + \frac{1}{48} \text{kurt}(y)^2 \tag{15}$$

The random variable y is assumed to be of zero mean and unit variance. To avoid the problems encountered with the above approximation of negentropy, new approximation were developed in [21]. These approximations were based on the maximum –entropy principal. In general we obtain the following approximation:

$$J(y) \approx \sum k_i [E\{G_i(y)\} - E\{G_i(\upsilon)\}]^2,$$

Where $k_i$ are some positive constants, and $\upsilon$ is a Gaussian variable of zero mean and unit variance (i.e., standardized). The variable y is assumed to be of zero mean and unit variance, and the functions $G_i$ are some nonquardratic functions [21]. Note that even in cases where this approximation is not very accurate, the above equation can be used to construct a measure of nongaussianity that is consistent in the sense that it is always non-negative, and equal to zero if $y$ has a Gaussian distribution.

In the case where we use only one nonquardratic function $G$. This is clearly a generalization of the moment-based approximation in (15), if $y$ is symmetric. Indeed taking $G(y) = y^4$, one then obtains exactly (15), i.e. a kurtosis-based approximation.

By choosing G wisely, one obtains approximations of negentropy that are much better than the one given by (15). In particular, choosing G that does not grow to fast, one obtains more robust estimators.

As example the following choices of G have proved very useful:

$$G_1(u) = \frac{1}{a_1} \log\cosh a_1 u, \quad G_2(u) = -\exp(-u^2/2)$$

Where $1 \le a_1 \le 2$ is some suitable constant.

Thus we obtain approximations of negentropy that give a very good compromise between the properties of the two classical nongaussianity measures given by kurtosis and negentropy.

## 2.10 Minimization of mutual information

Another approach for ICA estimation, inspired by information theory, is minimization of mutual information. By minimizing the mutual information we find that it leads to the same principal of finding most nongaussian directions as was stated above.

### 2.10.1 Mutual information

Using the concept of differential entropy, we define the mutual information I between m (scalar) random variables $y_i, i = 1 \cdots m$ as follows

$$I(y_1, y_2, \cdots, y_n) = \sum_{i=1}^{m} H(y_i) - H(y).$$

Mutual information is a natural measure of the dependence between random variables. In fact, it is equivalent to the well- known Kullback-Leibler divergence between the joint density $f(y)$ and the product of its marginal densities; a very natural measure of independence. It is always non-negative, and zero if and only if the variables are statistically independent. Thus mutual information takes in to account the whole dependence structure of the variables, and not only the covariance, like PCA and related methods.

Mutual information can be interpreted by using the interpretation of entropy as code length. The term $H(y_i)$ give the lengths of codes for the $y_i$ when these are coded separately, and $H(y)$ gives the code length when $y$ is coded as a random vector, i.e. all the components are coded in the same code. Mutual information thus shows what code length reduction is obtained by coding the whole vector instead of the separate components. In general, better codes can be obtained by coding the whole vector. However, if the $y_i$ are independent, they give no information on each other and one could just as well code the variables separately without increasing code length.

An important property of mutual information [12, 19] is that we have for an invertible linear transformation $y = Wx$:

$$I(y_1, y_2, \cdots, y_n) = \sum_i H(y_i) - H(x) - \log|\det W| \qquad (16)$$

Now let us consider what happens if we constrain the $y_i$ to be uncorrelated and of unit variance. This means $E\{yy^T\} = W\,\mathrm{E}\{xx^T\}W^T = \mathrm{I}$, which implies $\det \mathrm{I} = 1 = \left(\det W\,\mathrm{E}\{xx^T\}W^T\right) = (\det W)\left(\det \mathrm{E}\{xx^T\}\right)\left(\det W^T\right)$, and this implies that det $W$ must be constant, and the sign. Thus we obtain,

$$I(y_1, y_2, \cdots, y_n) = C - \sum_i J(y_i). \qquad (17)$$

Where $C$ is a constant that does not depend on $W$. This shows that the fundamental relation between negentropy and mutual information.

## 2.10.2 Defining ICA by mutual information

Since mutual information is the natural information-theoretic measure of the independence of random variables, we could use it as the criteria for finding the ICA transform. In this approach that is an alternative to the model estimation approach, we define the ICA of a random vector x as an invertible transformation as in (4) where the matrix W is determined so that the mutual information of the transformed components $s_s$ is minimized.

It is now obvious from (17) that finding an invertible transformation W that minimizes the mutual information is roughly equivalent to finding directions in which the negentropy is maximized. More precisely, it is roughly equivalent to finding 1-D subspaces such that the projections in those subspaces have maximum negentropy. Rigorously, speaking, (17) shows that ICA estimation by minimization of mutual information is equivalent to maximizing the sum of nongaussianities of the estimates, when the estimates are constrained to be uncorrelated. The constraint of uncorrelated ness is in fact not necessary, but simplifies the computations considerably, as one can then use the simpler form in (17) instead of the more complicated form in (16).

Thus, we see that the formulation of ICA as minimization of mutual information gives another rigorous justification of our more heuristically introduced idea of finding maximally nongaussian directions.

## 2.11 Maximum likelihood estimation

A very popular approach for estimating the ICA model is maximum likelihood estimation, which is closely connected to the Infomax principle. Here we will see that, it is essentially equivalent to minimization of mutual information.

It is possible to formulate directly the likelihood in the noise-free ICA model, which was done in [22], and then estimate the model by a maximum likelihood method. Denoting $W = (w_1, \cdots, w_n)^T$ the matrix $A^{-1}$, the log-likelihood takes the form [19].

$$L = \sum_{t=1}^{T} \sum_{i=1}^{n} \log f_i(w_i^T x(t)) + T \log|\det W| \tag{18}$$

Where the $f_i$ are the density function of the $s_i$ (here assumed to be known), and the $x(t), t = 1, \cdots, T$ are the realization of $x$. The term $\log|\det W|$ in the likelihood comes from the classic rule for (linearly) transforming random variables and their densities [12]. In general for any random vector x with density $p_x$and for any matrix W, the density of $y$ = $Wx$ is given by $p_x(Wx)|\det W|$.

## 2.12 The Infomax Principle

Another related contrast function was derived from a neural network viewpoint in [23, 24]. This was based on maximizing the output entropy (or information flow) of a neural network with non-linear outputs. Assume that x is the input to the neural network whose outputs are of the form $g_i(w_i^T x)$, where the $g_i$are the some non-linear scalar functions, and the $w_i$ are the weight vectors of the neurons. One then wants to maximize the entropy of the outputs:

$$L_2 = H(g_1(w_1 x), \cdots, g_n(w_n x)). \tag{19}$$

If the $g_i$ are well chosen, this framework also enables the estimation of the ICA model. Indeed, several authors, e.g. , [ 25,26], proved the surprising result that the principle of network entropy maximization, linearities $g_i$ used in the neural network are chosen as the cumulative distribution functions corresponding to the densities $f_i$, i.e., $g_i'(.) = f_i(.)$.

## 2.13 Connection to mutual information

To see the connection between likelihood and mutual information, consider the expectation of log likelihood:

$$\frac{1}{T}\mathrm{E}\{L\} = \sum_{i=1}^{n} \mathrm{E}\{\log f_i(w_i^T x)\} + \log|\det W| \tag{20}$$

Actually if the $f_i$ were equal to the actual distribution of $w_i^T x$, the first term would be equal to $-\sum_i H(w_i^T x)$.Thus the likelihood would be equal, up to an additive constant, to the negative of mutual information as given in eq. ( 17) .In practice the connection is even stronger. This is because in practice we do not know the distribution of the independent components. A reasonable approach to estimate the density of $w_i^T x$ as part of the ML estimation method, and use this as an approximation of the density $s_i$. In this case, likelihood and mutual information are, for all practical purposes, equivalent. Nevertheless, there is a small difference that may be very important in practice. The problem with maximum likelihood estimation is that the densities $f_i$ must be estimated correctly. They need not be estimated with any great precision: in fact it is enough to estimate whether they are sub- or supergaussian [25, 27, and 28]. In many cases, in fact, we have enough prior knowledge on the independent components, and we don't need to estimate their nature from the data. In any case, if the information on the nature of the independent components is not correct, ML estimation, will give completely wrong results. Some care must be taken with ML estimation, therefore. In contrast, using reasonable measures of nongaussianity, this problem does not usually arise.

## 2.14 ICA and Projection pursuit

Projection Pursuit [2, 3, 4, and 29] is a technique developed in statistics for finding "interesting" projections of multidimensional data. Such projections can then be used for optimal visualization of the data, and for such purposes as density estimation and regression. In basic (1-D) projection pursuit, we try to find directions such that the projections of the data in those directions have interesting distributions, i.e., display some structure. It has been argued by Huber [3] and by Jones and Sibson [4]. That the Gaussian distribution is the least interesting one, and that most interesting directions are those that show the least Gaussian distribution. This is exactly what we do to estimate the ICA model.

## 2.15 ICA Algorithms:

After choosing one of the principles of estimation for ICA, one needs a practical method for its implementation. Usually, this means that after choosing an objective (contrast) function for ICA, we need to decide how to optimize it.

### 2.15.1 Jutten-Hèrault algorithm

The pioneering work in [30] was inspired by neural networks. Their algorithm was based on canceling the non-linear cross-correlations. The non-diagonal terms of matrix $W$ are updated according to

$$\Delta W_{ij} \propto g_1(y_i)g_2(y_j), \text{ for } i \neq j \tag{21}$$

Where $g_1$ and $g_2$ are the sum odd non-linear functions, and the $y_i$ are computed at every iteration as $y = (I + W)^{-1}x$. The diagonal terms $W_{ii}$ are set to zero. The $y_i$ then give, after convergence, estimates of the independent components. Unfortunately, the algorithms converge only under rather severe restrictions [31].

## 2.15.2 Non-linear decorrelation algorithm

Further algorithms for canceling non-linear cross-correlations were introduced independently in [32, 33, and 34] and [35, 36]. Compared to the Jutten-Hêrault algorithm, these algorithms reduce the computational overhead by avoiding any matrix inversions, and improve its stability. Such as, the following algorithm was given in [33, 34]:

$$\Delta W \propto (I - g_1(y) g_2(y^T))W, \tag{22}$$

Where $y = Wx$, the non-linearities $g_1(.)$ and $g_2(.)$are applied separately on every component of the vector y, and the identity matrix could be replaced by any positive definite diagonal matrix. In [35, 36], the following algorithm called the EASI algorithm was introduced:

$$\Delta W \propto (I - yy^T - g(y)y^T + yg(y^T))W, \tag{23}$$

A principled way of choosing the non-linearities used in these learning rules is provided by the maximum likelihood (or Infomax) approach.

## 2.15.3 Non-linear PCA algorithm

Nonlinear extensions of the well-known neural PCA algorithms [37, 38, and 39] were developed in [40]. For example, in [40], the following non-linear version of a hierarchical PCA learning rule was introduced:

$$\Delta W_i \propto g(y_i)x - g(y_i)\sum_{j=1}^{i} g(y_i)w_j \tag{24}$$

Where g is a suitable non-linear scalar function. The symmetric versions of the learning rules in [38, 39] can be extended for the non-linear case in the same manner. In general, the introduction of non-linearities means that the learning rule uses higher-order representation techniques (projection pursuit, blind deconvolution, ICA).In [41, 42], it was proved that for well-chosen non-linearities, the learning rule in (24) does indeed perform ICA, if the data is sphered (whitened). Algorithms for exactly maximizing the nonlinear PCA criteria were introduced in [43].

An interesting algorithm simplification of the non-linear PCA algorithm is the bigradient algorithm [44]. The feedback term in the learning rule (24) is here replaced by much simpler one, giving

$$W(t+1) = W(t) + \mu(t)g(W(t)x(t))x(t)^T + \alpha(\mathrm{I} - W(t)W(t)^T)W(t) \qquad (25)$$

Where $\mu(t)$ is the learning rate (step size) sequence, $\alpha$ is a constant on the range (0.5 to 1), the function g is applied separately on every component of the vector $y = Wx$, and the data is assumed to be sphered. A hierarchical version of the bigradient algorithm is also possible. Due to the simplicity of the bigradient, its properties can be analyzed in more detail, as in [44] and [13].

### 2.15.4 Neural one-unit learning rules

Using the principle of stochastic gradient descent, one can derive simple algorithms from the one -unit contrast functions explained above. Let us consider first whitened data. For example, taking the instantaneous gradient of the generalized contrast function { $J(y) \approx \sum k_i[E\{G_i(y)\} - E\{G_i(\upsilon)\}]^2$ } with respect to w, and taking the normalization $\|w\|^2 = 1$ in to account, one obtains the following Hebbian-like learning rule

$$\Delta W \propto rxg(w^T x) \qquad (26)$$

Where constant may be defined, e.g. as $r = \mathrm{E}\{G(w^T x)\} - \mathrm{E}\{G(\upsilon)\}$. The nonlinearity g can thus be almost any nonlinear function;

### 2.15.5 Tensor-based algorithms

Algorithms utilizing the fourth-order cumulant tensor for estimation of ICA [45, 46, 47, 48, 49, 8]. These are typically batch algorithms (non-adaptive), using such tensorial techniques as eigenmatrix decomposition, which is a generalization of eigenvalue decomposition for higher-order tensors. Such decomposition can be performed using ordinary algorithms for eigen-value decomposition of matrices, but this requires matrices of size $m^2 \times m^2$. Since such matrices are often too large, specialized Lanczos type algorithms of lower complexity have also been developed [45]. These algorithms often perform very efficiently on small dimensions. However, in large dimensions, the memory requirements may be prohibitive, because often the coefficients of the 4-th order tensor must be stored in memory, which requires $O(m^4)$ units of memory. The algorithms also tend to be quite complicated to program.

## 2.15.6 Weighted covariance methods:

The eigenvalue decomposition of the weighted covariance matrix data allows the computation of the ICA estimates using standard methods of linear algebra [51] on matrices of reasonable complexity (m x m). Here, the data must be sphered. This method is computationally highly efficient, but, unfortunately, it works only under the rather severe restriction that the kurtosis of the independent components are all different.

## 2.15.7 The Fast ICA Algorithm

We have already seen the different measures of nongaussianity, i.e. objective functions for ICA estimation. In practice, one also needs an algorithm for maximizing the contrast function, for example the one in ( ). In this section we will elaborate a very efficient method of maximization suited for the task stated above. Here it is already assumed that the data is preprocessed by centering and whitening.

### 2.15.7.1 Fast ICA for one unit

In the one unit version of Fast ICA. By a "unit" we refer to a computational unit, an artificial neuron, having a weight vector $w$ that the neuron is able to update by a learning rule. The fast ICA learning rule finds a direction, i.e. a unit vector w such that the projection $w^T x$ maximizes nongaussianity. Nongaussianity is here measured by the approximation of negentropy

$J(w^T x)$ given in ( ). The variance of $w^T x$ must here be constrained to unity; for whitened data this is equivalent to constraining the norm of w to be unity.

The Fast ICA is based on a fixed-point iteration scheme for finding a maximum of the nongaussianity of $w^T x$, as measured in ( ), see [9, 21]. It can also be derived as an approximation Newton Iteration [21]. Denote by g the derivative of the nonquardratic function G used in ( ); for example the derivatives of the functions in ( ) are:

$$g_1(u) = \tanh(a_1 u)$$
$$g_2(u) = u\exp(-u^2/2) \qquad (21)$$

Where $1 \le a_1 \le 2$ is some suitable constant, often taken as $a_1 = 1$. The basic form of the Fast ICA algorithm is as follows:

1. Choose an initial (e.g. random) weight vector w.
2. Let $w^+ = \mathrm{E}\{xg(w^T x)\} - \mathrm{E}\{g'(w^T x)\}w$
3. Let $w = w^+ / \|w^+\|$
4. If not converged, go back to 2.

Convergence means that the old and new values of w point in the same direction, i.e. their dot-product are (almost) equal to 1. It is not necessary that the vector converges to a single point, since *w* and -*w* define the same direction. This is again because the independent components can be defined only up to a multiplicative sign. We assumed here that the data is already prewhitened.

The derivation of Fast ICA is as under

First note that the maxima of the approximation of the negentropy of $w^T x$ are obtained at certain optima of $\mathrm{E}\{G(w^T x)\}$. According to the Kuhn-Tucker conditions [17], the optima of $\mathrm{E}\{G(w^T x)\}$ under the constraint $\mathrm{E}\{G(w^T x)^2\} = \|w\|^2 = 1$ are obtained at points where

$$\mathrm{E}\{\mathrm{x}g(\mathrm{w}^{\mathrm{T}} x)\} - \beta w = 0 \qquad (22)$$

Let us try to solve this equation by Newton's method. Denoting the function on the left hand side of (25) by F, we obtain its Jacobean matrix *J F (w)* as

$$JF(w) = E\{xx^T g'(w^T x)\} - \beta I \qquad (23)$$

To simplify the inversion of this matrix, we decide to approximate the first term in (23).Since the data is sphered, a reasonable approximation seems to be

$$E\{xx^T g'(w^T x)\} \approx E\{xx^T\} E\{g'(w^T x)\} = E\{g'(w^T x)\} I \qquad (24)$$

Thus the Jacobean matrix becomes diagonal, and can easily be inverted. Thus we obtain the following approximative Newton iteration:

$$w^+ = w - [E\{xg(w^T x)\} - \beta w]/[E\{g'(w^T x)\} - \beta] \qquad (25)$$

This algorithm can be further simplified by multiplying both sides of (25) by $\beta - E\{g'(w^T x)\}$. This gives, after algebraic simplification, the Fast ICA iteration.

In practice, the expectation in Fast ICA must be replaced by their estimates. The natural estimates are of course the corresponding sample means. Ideally, all the data available should be used, but this is often not a good idea because the computations may become too demanding. Then the averages can be estimated using a smaller sample, whose size has a considerable effect on the accuracy of the final estimates. The sample points should be chosen separately at every iteration. If the convergence is not satisfactory, one may then increase the sample size.

### 2.15.7.2 Fast ICA for several units

The one-unit algorithm of the preceding subsections just one of the independent components, or one projection pursuit direction. To estimate several independent components, we need to run the one –unit Fast ICA algorithm using several units (e.g. neurons) with weight vectors $w_1 \cdots, w_n$.

To prevent different vectors from converging to the same maxima we must decorrelate the outputs $w_1^T x, \cdots, w_n^T x$ after every iteration. Methods for achieving this are as under.

A simple way of achieving decorrelation is a deflation scheme based on a Gram-Schmidt-like decorrelation. This means that we estimate the independent components one by one. When we have estimated p independent components, or p vectors $w_1, \cdots, w_p$, we run the one -unit fixed point algorithm for $w_{p+1}$, and after every iteration step subtract from $w_{p+1}$ the "projections" $w_{p+1}^T w_j w_j, j = 1, \cdots, p$ of the previously estimated p vectors, and then renormalize $w_{p+1}$:

1. Let $w_{p+1} = w_{p+1} - \sum_{j=1}^{p} w_{p+1}^T w_j w_j$

2. Let $w_{p+1} = w_{p+1} \Big/ \sqrt{w_{p+1}^T w_{p+1}}$ (26)

In certain applications, however, it may be desired to use a symmetric decorrelation, in which no vectors are "privileged" over others [41]. This can be accomplished, e.g., by the classical method involving matrix square roots.

$$\text{Let } W = \left(WW^T\right)^{-1/2} W \qquad (27)$$

Where W is the matrix $(w_1, \cdots, w_n)^T$ of the vectors, and the inverse square root $\left(WW^T\right) = \mathrm{F} \wedge \mathrm{F}^{\mathrm{T}}$ as $\left(WW^T\right)^{-1/2} = \mathrm{F} \wedge^{-1/2} \mathrm{F}^{\mathrm{T}}$.

A simple alternative is the following iterative algorithm [51],

1. Let $W = W \Big/ \left\| WW^T \right\|$

2. Let $W = \frac{3}{2} W - \frac{1}{2} WW^T W$ (28)

Repeat 2. Until convergence:

The norm in step 1 can be any ordinary matrix norm, e.g., the 2-norm or the largest absolute row (or column) sum.

### 2.15.7.3 Fast ICA and maximum likelihood

We will see a version of Fast ICA that shows explicitly the connection to the well-known infomax or maximum likelihood algorithm introduced in [52, 53, 25, and 36]. If we express Fast ICA using the intermediate formula in (25), and write it in matrix form [21], we see that Fast ICA takes the following form:

$$W^{+} = W + \Gamma[\mathrm{diag}(-\beta_{i}) + \mathrm{E}\{g(y)y^{T}\}]W \tag{29}$$

Where $y = Wx$, $\beta_i = \mathrm{E}\{y_i g(y_i)\}$, and $\Gamma = \mathrm{diag}\left( 1/(\beta_i - \mathrm{E}\{g'(y)\}) \right)$. The matrix $W$ needs to be orthogonalized after every step. In this matrix version, it is natural to orthogonalize $W$ symmetrically.

The above version of Fast ICA could be compared with the stochastic gradient method for maximizing likelihood [52, 23, 16, and 27]:

$$W^{+} = W + \mu[\mathrm{I} + g(y)y^{T}]W. \tag{30}$$

Where $\mu$ is the learning rate, not necessarily constant in time. Comparing (29) and (30), we see that Fast ICA can be considered as a fixed-point algorithm for maximum likelihood estimation of the ICA data model. In Fast ICA, convergence speed is optimized by the choice of the matrices $\Gamma$ and diag($-\beta_i$).Another advantage of Fast ICA is that it can estimate both sub- and super-Gaussian independent components, which is contrast to ordinary ML algorithms, which only works for a given class of distribution.

### 2.15.7.4 Properties of the Fast-ICA Algorithm

The Fast ICA algorithm and the underlying contrast functions have a number of desirable properties when compared with existing methods for ICA. The convergence is cubic (or at least quadratic), under the assumption of the ICA data model [54]. This is in contrast to ordinary ICA algorithms based on (stochastic) gradient descent methods, where the convergence is only linear. This means a very fast convergence, as has been confirmed by simulations and experiments on real data [55].

1. Contrary to gradient-based algorithms, there are no step size parameters to choose. This means that the algorithm is easy to use.
2. The algorithm finds directly independent components of (practically) any non-Gaussian distribution using any nonlinearity g. This is in contrast to many algorithms, where some estimates of the probability distribution function has to be first available, and the nonlinearity must be chosen accordingly.
3. The performance of the method can be optimized by choosing a suitable nonlinearity g. In particular, one can obtain algorithms that are robust and of minimum variance. In fact, the two nonlinearities in (21) have some optimal properties [54].
4. The independent components can be estimated one by one, which is roughly equivalent to doing projection pursuit. This is useful in exploratory data analysis, and decreases the computational load of the method in cases where only some of the independent components need to be estimated.
5. The Fast ICA has most of the advantages of neural algorithms: It is parallel, distributed, computationally simple, and requires less memory space. Stochastic gradient methods seem to be preferable only if fast adaptivity in a changing environment is required.

## 2.15.8 HO-ICA Algorithm (Proposed By Us)

### 2.15.8.1 HO-ICA Model

This is the new single neuron based Independent Components separation technique. In this model we use the higher order statistics for the observed by the sensors. In mathematical sense we can say that if one have two observed signals $x_1$ and $x_2$. We can use $x_1, x_2, x_1 * x_2$, $x_1^2$ , $x_2^2$, $x_1^2 * x_2^2$, any order or any combination of the signals as the inputs of our algorithm. The unmixing matrix in this case will not be square as in the other algorithm it is a necessary condition that unmixing matrix should be a square matrix.

### 2.15.8.2 HO-ICA Algorithm:

1. Choose a random initial weight vector $\mathbf{w}$.
2. $$\mathbf{w} \;=\mathbf{w}+\eta * d\mathbf{w}$$
   $$d\mathbf{w}=(I+(1-2*\log sig(y))*y'))*w$$
   $$\mathbf{w}=\mathbf{w}/\,absolute\|\mathbf{w}\|$$
3. Repeat until the $\mathbf{w}$ converges.

## 2.16 Comparison of the HO-ICA Algorithm with all other existing Algorithms

Till date all the researchers have sown that the Fast-ICA technique based on the fixed point algorithm with hyperbolic tangent non-linearity is the most suitable, fast and giving the expected results. There is only one MatLab- based package of Fast-ICA, is available at the laboratory of Computer & Information Science Finland. We have compared our proposed algorithm with that one.

| S. No. | Fast-ICA Algorithm | HO-ICA Algorithm |
|---|---|---|
| 1. | It works up to the maximum 4th order of inputs. | Works for any order any combination of the inputs. |
| 2. | A large no. of pre-processing and post-processing required for the image and audio separation. | Comparatively nominal pre-processing and post-processing required |
| 3. | Deals with only the linear input data | Deals with all linear as well as non-linear combination of the data. |
| 4. | Slow | Very much Fast |
| 5. | It works only when the no. of sensor outputs are equal to the sensor inputs | Works for all the input, output combinations such as less, equal or greater. |

## 2.17 Applications of ICA

The most classical application of ICA, the cocktail-party problem.

### 2.17.1 Separation of Artifacts in MEG Data

Magneto encephalography (MEG) is a noninvasive technique by which the activity or the cortical neurons can be measured with very good temporal resolution and moderate spatial resolution. When using a MEG record, as a research or clinical tool, the investigator may face a problem of extracting the essential features of the neuromagnetic signals in the presence of artifacts. The amplitude of the disturbances may be higher than the brain signals, and artifacts may resemble pathological signals in shape.

In [56], the author has introduced a new method to separate brain activity from artifacts using ICA. The approach is based on the assumption that the brain activity and the artifacts, e.g. eye movements are blinks, or sensor malfunctions, are automatically and physiologically separate processes, and this separation is reflected in the statistical independence between the magnetic signals generated by those processes.

### 2.17.2 Finding Hidden Factors in Financial Data

It is a tempting alternative to try ICA on financial data. There are many situations in that application domain in which parallel time series are available, such as currency exchange rates or daily returns of stocks, that may have some common underlying factors. ICA might reveal some driving mechanisms that otherwise remain hidden. In a recent study of stock portfolio [53], it was found that ICA is a complimentary tool to PCA, allowing the underlying structure of the data to be more easily observed. For example: the cash flow of several stores belonging to the same retail chain, Trying to find the fundamental factors common to all stores that affect the cash flow data. Thus the cash flow effect of the factors specific to any particular store, i.e., the effect of the actions taken at the individual store and in its local environment could be analyzed.

The assumptions of having some underlying independent components in this specific application may not be unrealistic. For example, factors like seasonal variations due to holidays and annual variations, and factors having a sudden effect on the purchasing power of the customers like prize changes of various commodities, can be expected to have an effect on all retail stores, and such factors can be assumed to like e.g. advertising efforts, the effect of the factors on the cash flow of specific retail outlets are slightly different. By ICA, it is possible to isolate both the underlying factors and the effect weights, thus also making it possible to group the stores on the basis of their managerial policies using only the cash flow time series data.

## 2.17.3 Reducing Noise in Natural Images

By finding the ICA filters for natural images and, based on the ICA decomposition, removing noise from images corrupted with additive Gaussian noise.

For the image denote the vector of pixel gray levels in an image window by x. Elements of the x are the location in image window or patch. We choose the sample windows at random locations, row by row scanning is used to turn a square image window in to a pixel values. The independent components of such image windows are represented in the under given fig. [6]. Each window in this figure corresponds to one of the columns of $a_i$ of the mixing matrix A. Thus an observed window is the superposition of these windows with independent coefficients. Now, suppose a noisy image model holds:

$$z = x + n$$

where n is uncorrelated noise, with elements indexed in the image window in the same way as x, and z is the measured image window corrupted with noise. Let us further assume that n is Gaussian and x is non-Gaussian. There are many ways to clean the noise.

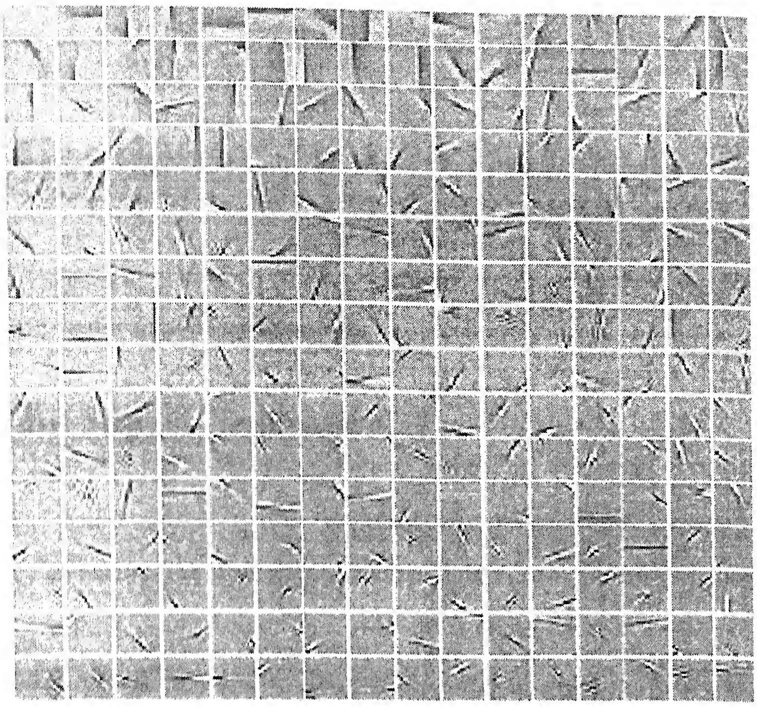


Figure 6. Basis functions in ICA of natural images. The input window size was $16 \times 16$ pixels can be considered as the independent features of images

## 2.17.4 Telecommunication

Another emerging application area of great potential: telecommunications. An example of a real - world communications application where blind source separation techniques are useful is the separation of the user's own signal from the interfering other user's signals in CDMA (Code-Division Multiple Access) mobile communications [57]. This problem is semi-blind in the sense that certain additional prior information is available on the CDMA data model. But the number of parameters to be estimated is often so high that suitable blind source separation techniques taking in to account the available prior knowledge provide a dear performance improvement over more traditional estimation techniques [57].

# CHAPTER-3

# Application of ICA to Different Fields

There are various applications of ICA discussed as above in the chapter 2. I have worked on the some under listed applications.

## 3.1: Blind Source Separation

As stated above in the chapter-2, that "The Blind Source Separation (BSS) problem is to extract the underlying source signals from the set of linear mixtures, where the mixing matrix is unknown. This problem is common in acoustics, radio, and medical signal, image processing and hyper spectral imaging, etc..," I have applied the HO- ICA algorithm for unmixing of the images. Andreas Jung [58] has discussed about the fast ICA algorithm for unmixing of images in his research work.

### 3.1.1: UNMIXING OF IMAGES:

Like unmixing of unknown signals, for proper visualization of three black and white images. To give a better impression, how the HO- ICA algorithm works, I have demonstrated it by demixing process of three mixed black and white images.

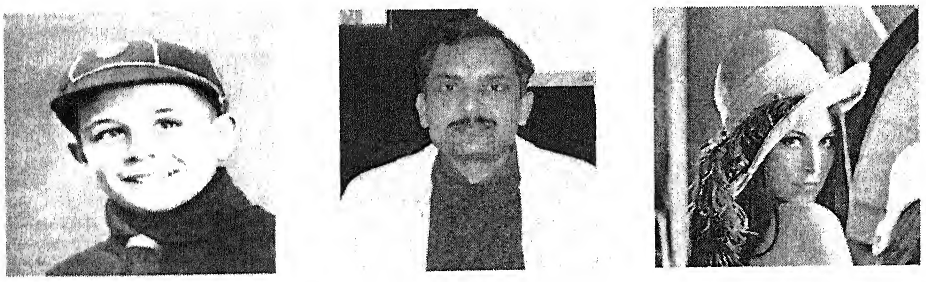


Fig.6 The original sources/ images exists of the first one is of 256×256, and second one is of 225×216, the third one is of 512×512= pixels with the resolution of 256 gray level.

The original images in the figure are of 256×256, 225×216 and 512×512 pixels with the resolution of 256 gray scales. Let us denote them as s; where the s matrix is of 3 rows and the no. of columns is 512×512, each row denotes the one black and white image. I have mixed these images, so that the mixed signals are recovered by x = As. We know the mixing matrix A. Now we got the two mixtures means here the condition is of as stated above in the second chapter of under-determined ICA, where sensors or outputs are less than the no. sources. So finally we got the mixture matrix x having two rows. In [58] Andreas Jung has taken the no. of sensors (outputs) equal to the no. of sensors.

Fig.7 Two mixtures of the above three black and white images of fig.6 (here the number of outputs are less than the inputs)

When representing these mixed images to an ICA- algorithm, the algorithm tries to unmix them by using only the property of independents of the original images. As we know that all the algorithms work iteratively so there is a place to introduce the new algorithm. We have introduced the HO- ICA algorithm as discussed above in the chapter-2 for unmixing or recovering the original images from the mixed images as shown under in the fig.8.

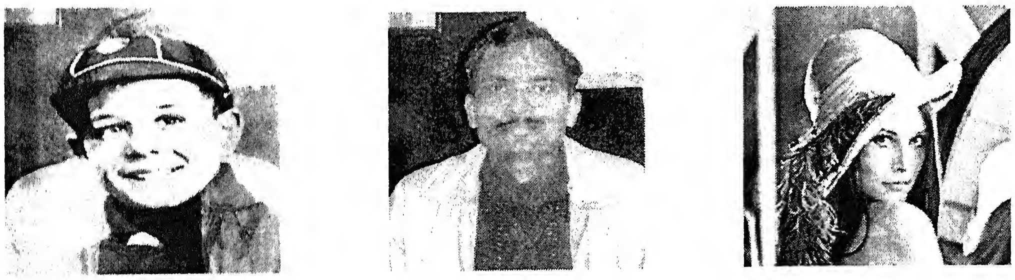


Fig.8 Recovered images from the mixed images by HO- ICA Algorithm

In the research work of Andreas Jung [58], He has introduced the images of same pixel values and restricted that the images which they have mixed are of size

250×250=62500 pixels with the resolution of 256 gray values, here we have taken the images of different pixel values, as the first one is of 256×256, the second one is of 225×216 and the third one is of 512×512 pixels. So we have eliminated two restrictions such as:

1. No. of sources equal to the no. of sensors.
2. Images of same pixel values.

Finally by seeing the recovered images in fig.8, one can correlate that the recovered images are nearly same as the source images.

## 3.2: Number Plate & Hidden Face detection

Suppose in hit and run case, if any vehicle meets with some accident and after that driver runs away with the vehicle. Cameras fitted over there take the pictures of the driver and the number plate, but the problem is that due to wind screen, picture of the driver does not come clear. One may not recognize the person driving the vehicle. In the case of number plate with problems like dust or improper angle of the focus area of the camera, the picture may not be clear. We have taken some real life pictures of the person driving the car and the number plate and applied HO-ICA algorithm to extract the observed face and the invisible data on the number plate. All the results extracted by the proposed algorithm for number plate and observed face are listed in the appendix A. Fig. 7 (a) is original image of number plate, fig. 7 (b,c,d,e,f,g,h) are of the extracted by the proposed algorithm. Fig. 8 (a) is original image of the person driving the car and fig. 8 (b, c, d, e, f) are the extracted faces. By seeing the results one can easily recognize the face and read the number plate.

## 3.3: Image Feature extraction

There are several goals in the image processing to apply ICA to image data. At least the objectives listed under can be identified.

1. Verify/ asses the applicability of the linear ICA model for image data and consider the limitations of the model.
2. Asses the applicability of the algorithm to the problem at hand.

3. Obtain the basic knowledge about the characteristics of the "independent components" of the data.

4. Consider the applicability of the results to various image processing tasks for example, Image Compression, Image Feature Extraction etc...

5. Interpret the results to asses the connections between ICA and other areas of research and the implications of these similarities and differences.

For starting of the process first of all we have to discuss about the preprocessing steps or the different parameters we have to choose. The parameters are as listed under.

1. Image data.
2. Sampling.
3. Data preprocessing.
4. Algorithm to apply and its parameters.
5. Number of sources or dimensionality reduction.
6. Estimation of statistical quantities or number of samples.

## 3.3.1: Image data

For getting the feature or independent components of the images, we have taken the 10 grayscale and 10 colored images, the sizes of the images are 256×256, 256×512, 512×512 or 512×256 pixels. The visual system has to operate in many different environments, so comparisons with it must also be based on the general data.

## 3.3.2: Sampling

In sampling we have to choose the size of the samples (sub images) we are taking from the images and the window function with which the window function taken.

The size of the window should be a compromise between two requirements.

1. It is large enough to contain sensible visual information even though the image is digitized and to retain the spectral properties of the original image.

2. It is small enough to introduce generality in to the data. To find any general properties in images, we have to restrain to local neighborhoods.

We did all the experiments for the windows of sizes 8×8, 16×16, 32×32 etc.

The most usual windowing method is the rectangular one, where a sample window of size N×N with top corner positioned at $x_0, y_0$ is obtained by multiplying the original image function $i(x, y)$ with window function

$$w(x, y, x_0, y_0) = \begin{cases} 1 & \text{if } x_0 \le x \le x_0 + N-1 \text{ and } y_0 \le y \le y_0 + N-1 \\ 0 & \text{Otherwise} \end{cases}$$

After the multiplication the area outside the window is discarded, leaving an $N \times N$ image window.

All the images which I have taken for the sampling are in appendix B.

### 3.3.3: Data Preprocessing

ICA algorithms require certain properties from the data to function correctly. Usually they require that the data be zero mean and whitened. We have already discussed about the whitening in chapter-2. In this section we will see the other preprocessing steps and their involvements or effects. What we are interested in images are local spatial properties like edges, corners, points, or lines. Because these correspond to changes in grayscale values, the local mean is subtracted from each sample to remove the effect of the uninteresting constant part.

Subtracting the local mean gives a new data vector

$$x_{zero,i} = x_{obs,i} - \frac{1}{m}\sum_{j=1}^{m} x_{obs,j} \quad , \quad i = 1,\cdots,m.$$

We can also write this equation as

$$x_{zero} = x_{obs} - \frac{1}{m} x^T I',$$

Where $I' = [1 1 \cdots 1]^T$

After subtraction of the local mean

$$E\{x_{zero,i}\} = E\{x_{obs,i} - \frac{1}{m}\sum_{j=1}^{m} x_{obs,j}\} = E\{x_{obs,i}\} - \frac{1}{m}E\{\sum_{j=1}^{m} x_{obs,j}\} = 0$$

So $E\{x_{zero}\} = 0$.

The algorithms used to extract ICA properties use extensively projections of samples onto ICA basis vectors. If some sample vectors are much longer than the others, their projections are significantly larger and they contribute much more to the result. This shows that image part with greater local variance have greater effect of the outcome of the algorithms, which is not what we want. This is why we equalize the local variance by normalizing the input variable $x_{zero}$, giving us

$$x_{norm} = \frac{x_{zero}}{\|x_{zero}\|}.$$

After this all the samples have an equal length.

### 3.3.4 Algorithms for the image feature extraction and its parameters

There are different algorithms for the image feature extraction Such as

1. Fixed-Point algorithm
2. JADE
3. The Bigradient algorithm
4. HO- ICA algorithm.

The models in which we have used the Fixed-Point algorithm and Bigradient algorithm are listed as under.

### 3.3.4.1 Large- Standard ICA simple-sell model

In this model we have used the parameters and algorithm is as under

1. Model = ICA (with FastICA algorithm, tanh nonlinearity)
2. Algorithm = Fixed-Point algorithm.
3. No. of independent components = 160.
4. No. of iterations = 500.

### 3.3.4.2 Large – Independent Subspace Analysis (complex cell model)

1. Model ISA (gradient descent with adaptive step size)
2. Algorithm = Gradient
3. No. of independent components = 64.
4. Step size = 0.1.
5. Convergence Parameter = 0.005.

### 3.3.4.3 Large – Topographic ICA (model for complex cells and topography)

1. Model = TICA (gradient descent with adaptive step size)
2. Algorithm = Gradient.
3. No. of Rows = 16.
4. No. of Columns = 10.

5. Step size = 0.1.

6. Convergence Parameter = 0.005.

#### 3.3.4.4 Small – Standard ICA (simple- cell model)

1. Model = ICA (with FastICA algorithm, tanh nonlinearity)

2. Algorithm = Fixed-Point algorithm.

3. No. of independent components = 64.

4. No. of iterations = 500.

#### 3.3.4.5 Small – Independent Subspace Analysis (complex cell model)

1. Model = ISA (gradient descent with adaptive step size)
2. Algorithm = Gradient
3. No. of independent components = 64.
4. Step size = 0.1.
5. Convergence Parameter = 0.005.

#### 3.3.4.6 Small – Topographic ICA (model for complex cells and topography)

1. Model = TICA (gradient descent with adaptive step size)

2. Algorithm = Gradient.

3. No. of Rows = 16.

4. No. of Columns = 10.

5. Step size = 0.1.

6. Convergence Parameter = 0.005.

### 3.3.4.7 HO- ICA Algorithms which we have proposed to extract the Image Features or Basis Vectors

Parameters which we have used for the HO- ICA algorithm are listed as under

1. Model = HO- ICA.
2. Algorithm = Higher- Order ICA Algorithm.
3. Epsilon = 1.5.
4. Convergence Parameter = 0.00001.
5. Activation Function = tanh(y).
6. W = randomly generated weights.
7. No. of independent components = 160.

The nonlinearities we have used with all the algorithms are *g(y) = tanh(y)* and $g(y) = \frac{1}{(1+e^{-y})}$.

To examine the dependence of the results on initialization we did the practices with the different starting points. The convergence parameter ∈ is chosen to be as small as possible, but large enough to allow for some errors in the method of estimation of statistical quantities involved and fluctuation in the numerical algorithm. This means that the smallest ∈ with which the algorithm seems to ensure convergence is found. The maximum no. of iterations chosen randomly.

JADE has only one parameter, which is used to determine when the diagnolization has converged. This is chosen automatically in the algorithm written by Cardoso, and the chosen value is based on the no. of available samples.

The bigradient algorithm is the one in which the problems occurred when parameters are concerned. It has two parameters.

1. The gradient descent / ascent step size $\mu$ advised in [59] that should be a small constant or decrease slowly with the no. of iterations. We have to use a small constant, since otherwise we would have to determine he descent rule for the parameter as well.

2. The Penalty parameter $\eta$ should be in between 0.5 to 1.

### 3.3.5 Number of sources or dimensionality reduction:

A constant window size we are using here, the number of "independent" sources that we are trying to find is nonconstant. If we have simple data, this data may not effectively span all directions in the space. (That is, the covariance matrix may have some very small but nonzero eigenvalues.) Fortunately PCA can be used to discover such situations.

### 3.3.6 Estimation of statistical quantities or number of samples:

The statistical quantities that have to be estimated in the algorithm are fortunately very simple expectations. So forming the estimates is no problem. The only remaining problem is the amount of data needed for these estimates. From 1to 3 we have used no. of samples = 1000 and the no. of independent components = 160 and size of the window is 16 by 16. From 5 to 7 we have chosen the no. of samples = 10000, no. of independent components = 64, window size = 8 by 8. In the our proposed algorithm we have chosen the window size of dimensions 16 by 16 as well as 8 by 8 and given the results.

## 3.4 Assessing the Results:

Since the work is highly an experimental research work, we should consider the ways in which we can asses the results.

1. The evaluation of the applicability of the algorithms is based partly on the results of the previous step and mostly on observing the behavior of the algorithms and

the visual quality of the results. For example, failure to converge is a clear indication of the unsuitability of the algorithm to this problem.

2. Basic Knowledge about the components is obtained mainly by visual inspection.

All the results obtained by different existing algorithm and the proposed HO- ICA algorithm we have shown in the Appendix B.

## 3.5 Image Compression:

Compressing an image is significantly different than the compressing raw binary data. Of course, general purpose comparison programs can be used to compress the images, but the results are less than the optimal. This is because images have certain statistical properties which can be exploited by encoders specifically designed for them.

Also, some of the finer details in the image can be sacrificed for the sake of saving little more bandwidth or storage space. This also means that lossy compression technique can be used in this area. Lossless compression involves with compressing data which when decompressed, will be an exact replica of the original data. This in the case when binary data such as executables, documents etc. are compressed. They need to be exactly reproduced when decompressed.

An approximation of the original image is enough for most purposes, as long as the error between the original and the compressed image is tolerable.

### 3.5.1 Measurement of Error

Two of the error matrices are used to compare the various image compression techniques are as Mean Square Error (MSE), and the Peak Signal to Noise Ratio (PSNR). The (MSE) is the cumulative squared error between the compressed and the original image, where (PSNR) is the measure of peak error.

The mathematical formulae for the two are as:

$$MSE = \frac{1}{MN}\sum_{y=1}^{M}\sum_{x=1}^{N}[I(x,y) - I'(x,y)]^2$$

$$PSNR = -20 * \log_{10}\left(\frac{MSE}{S^2}\right).$$

Where $I(x, y)$ is the original image and $I'(x, y)$ is the approximated version of the decompressed image, M &N are the dimensions of the image. A lower value of the MSE means lesser error, and as the inverse relationship with the PSNR, this translates that the high value of the PSNR.

Logically, a higher value of PSNR is good because it means that the ratio of signal to noise ratio is higher. Here the signal is the original image and the noise is the error in reconstruction. So the conclusion is, if a compression technique having a lower MSE (and a high PSNR) is the better one.

## 3.5.2 Image Compression Algorithms:

There are some algorithms available for the image compressions are as under.

1. Wavelet Compression.
2. JPEG/DCT Compression.
3. Vector Quantization Techniques (VQT).
4. Fractal Compression.
5. HO- ICA Compression (Proposed by us).

All the four algorithms are discussed in detail in [60, 61]. In [60], it is shown by comparing all of them that the wavelet compression based on the concept of the embedded zero tree wavelet transform on an image has significantly larger PSNR values and a better visual quality of decoded images compared with the other approaches, at a desired compression of 0.25 bits per pixel (bpp). Over the past few years, a variety of novel and sophisticated wavelet-based image coding schemes have been developed. These include EZW, SPIHT, SFQ, CREW, EPWIC, EBCOT, SR, Second Generation Image Coding, Image Coding using Wavelet Packets, Wavelet Image Coding using VQ, and Lossless Image Compression using Integer Lifting. In [61], it has been mentioned that JPEG-2000 based on the DWT in place of DCT is the real image compressor and better than the other compression techniques. We have compressed the image by applying the HO- ICA algorithm in place of the DWT and compared the results with the JPEG-2000 compressor.

All the results we have shown in the appendix C.

### 3.5.3 Image Compression by the HO - ICA algorithm

We have taken the images of Lena and the image of boy in grayscale and compressed them by increasing the order of the algorithm. All the results and size of the Lena image we have shown in table-1.

**TABLE-1**

| S. No. | Size on the disk in (KB) | PSNR in (dB) | Order of the Algorithm |
|---|---|---|---|
| 1 | 35.2 | 64.786 | 1st |
| 2 | 32.0 | 60.7402 | 2nd |
| 3 | 27.2 | 52.8746 | 3rd |
| 4 | 23.0 | 40.8746 | 4th |
| 5 | 19.4 | 22.6423 | 5th |
| 6 | 16.6 | 21.5831 | 6th |
| 7 | 12.7 | 20.2038 | 7th |
| 8 | 10.2 | 18.8838 | 8th |
| 9 | 8.61 | 16.8413 | 9th |
| 10 | 6.47 | 15.5831 | 10th |
| 11 | 4.43 | 14.2038 | 11th |

By seeing the size of the images and PSNR values we are achieving the good results up to the 8th order, but after that when we increase the order image quality starts deteriorating. So we have to compromise with the results as well as the quality.

### 3.5.4 Experimental Comparison

We have compressed the same images by JPEG-2000 compressor and by HO-ICA algorithm and compared the results that are listed in table-2.

TABLE-2

| Algorithm | PSNR Values in (dB) | | | |
|---|---|---|---|---|
| | Lenna | Boy | Fingerprint | kk |
| JPEG-2000 with DWT | 65.7316 | 71.328 | 63.45 | 62.65 |
| HO- ICA | 60.6283 | 68.382 | 61.23 | 62.0 |

## 3.6 Colored Image Compression:

In colored image compression, break the RGB components and applied the HO - ICA algorithm for the each component separately and then added them up. Results of final compressed images are in the appendix C.

# CHAPTER-4

# Conclusion and Future Scope

## 4.1 Conclusion

This thesis has sought to extend ICA and make it more powerful, flexible and more widely applicable. By implementing HO-ICA algorithm we have come out of the limitations related to the order of statistics, because all the algorithms implemented so far are unable to work out when the order increases beyond 4. After presenting some general theory in chapter -1 & chapter-2, we have applied HO-ICA algorithm in the blind source separation, number plate and hidden face detection, image feature extraction, image compression. In the case of blind source separation we have discussed about the unmixing of 3 images and compared the results with the fast-ICA algorithm. The results, when the number of sensor outputs is less than the inputs, have been shown.

After this, we have applied HO-ICA algorithm for image feature extraction. The image data consists of 13 images, from which a large set of 16×16 sub images were extracted and used as the mixed data. We first applied the existing algorithms. The first one is based on diagnolization of cumulant matrices and the second one is a gradient descent algorithm for optimizing a contrast function. The third algorithm type is a class of fixed point algorithms for optimizing a contrast function. We have reduced the dimension of the data from 160 to 64 by applying the algorithms (existed and the HO-ICA). The purpose of reduction of dimensionality is to suppress noise and avoid some of the problems associated with the high dimensional data. As shown in the results, the algorithms are able to extract meaningful features from the data. These features included lines, bars, edges, spots, and larger low frequency structures.

The results of the fixed-point & the HO-ICA algorithms with the hyperbolic tangent nonlinearity are able to sustain their ability to find the meaningful features from the data. On the other hand, the results from existing algorithms exhibited partial or unclear shapes.

In the case of image compression we have performed the experimental comparison on the gray scale images by applying proposed algorithm and one of the existing algorithms available in image compressing software JPEG-2000 with DWT. More or less similar performance has been found in these results. In the case of colored images we have compressed the image of Spiderman with the proposed algorithm and the algorithms in the softwares JPEG-2000 with DWT, ReaCompressor and Wavelet compressor and compared the results. Images compressed by JPEG-2000 compressor and our algorithm are competitive and preserve better quality than others.

## 4.2 Future Scope

In this thesis work we have shown the unmixing of images when the number of sources is known. There is still an open challenge for the researchers when the number of sources is unknown.

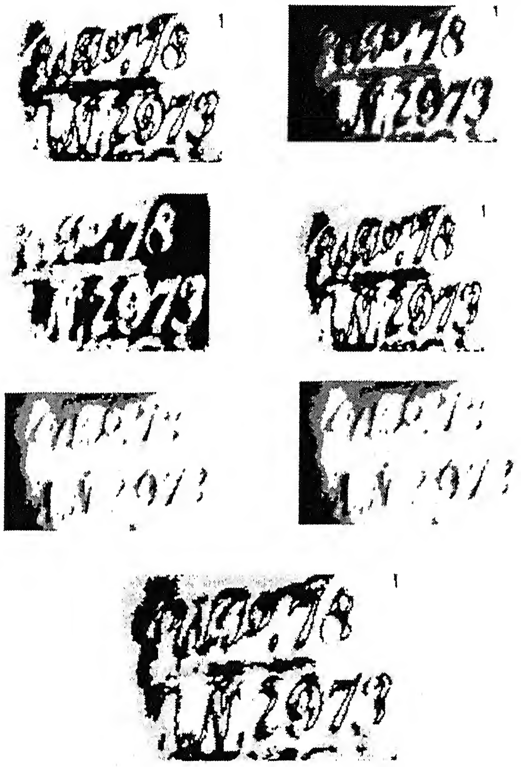
In the case of image feature extraction we have extracted either 160 or 64 independent components of the 13 images. There is still a scope of further research to reconstruct the original images after getting these components.

For number plate and hidden face extraction there is still a scope of research for getting the images of better quality.

In the field of image compression one can develop one's own codec to save image in the format that is useful to reduce the size of image preserving the better quality, like in wavelet compression wlt format and in JPEG-2000 jpg format.

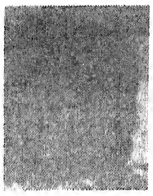
## APPENDIX A

# Number-plate Recognitior

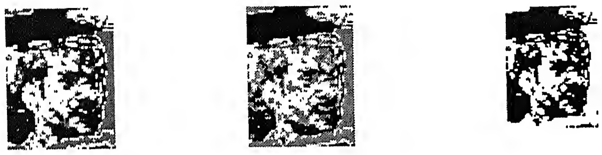


Recognize last digit.
Is it 3 or 8?

## Hidden Face detection:



(Observed Face)



(Extracted Faces by Applying HO-ICA Algorithm)

# APPENDIX B

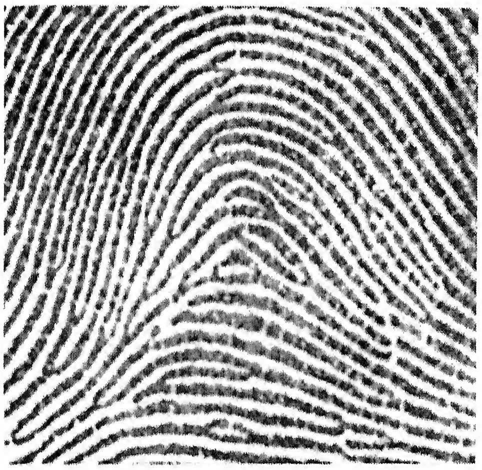


Fig. 01.jpg

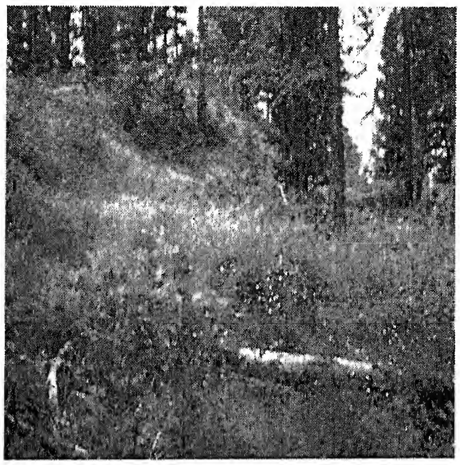


Fig. 02. jpg

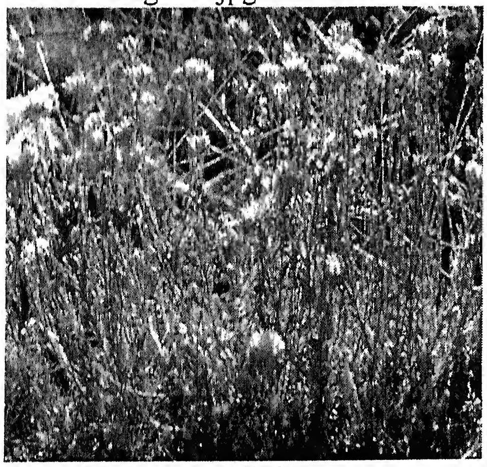


Fig. 03.jpg

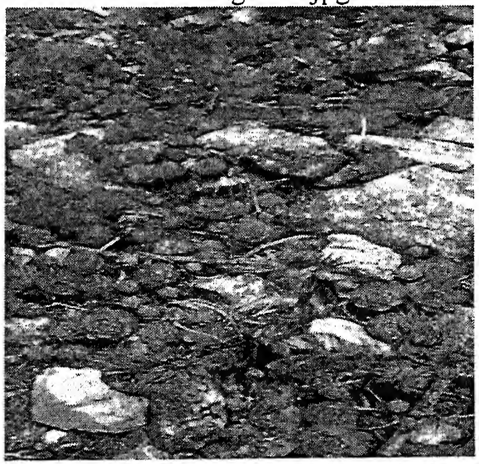


Fig. 04.jpg

Fig. 05.jpg

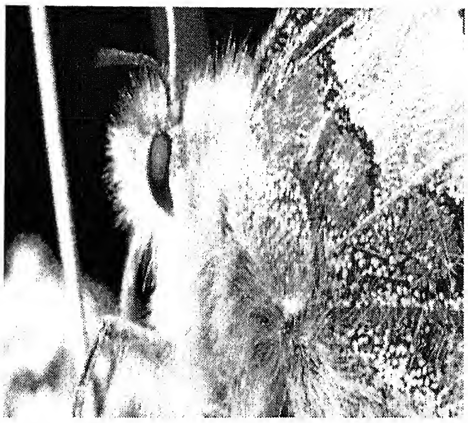


Fig. 06. jpg

Fig. 07. jpg

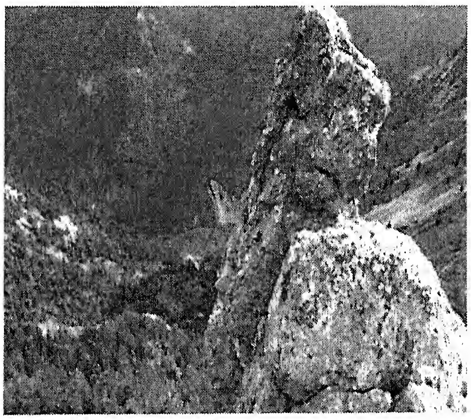


Fig. 08.jpg

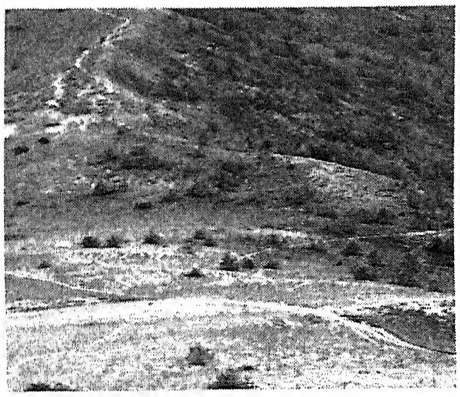


Fig. 09.jpg

Fig. 10.jpg

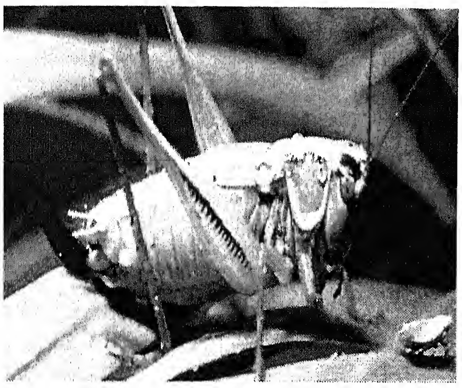


Fig. 11.jpg

Fig. 12.jpg

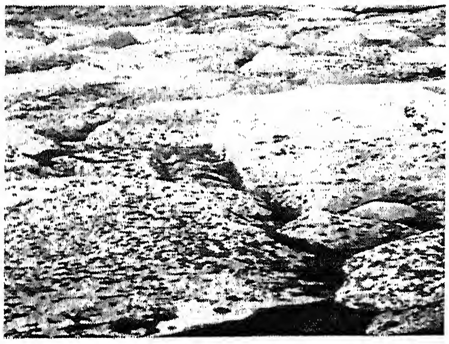
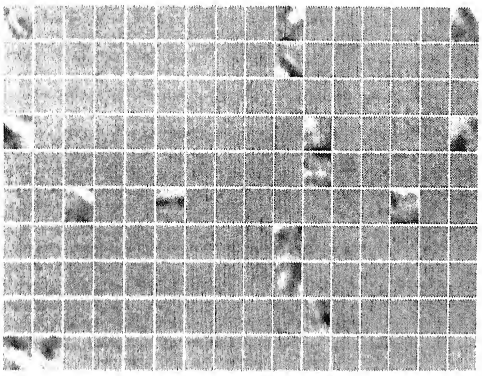
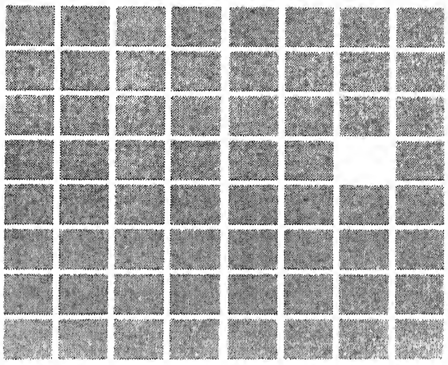


Fig. 13.jpg

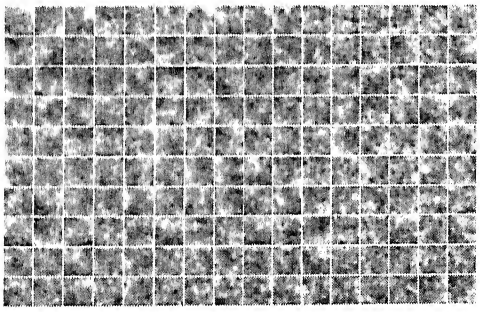
## Independent features (Basis Vectors) of 13 images shown above



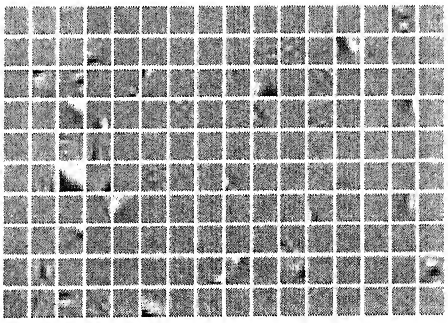
(ICA basis vectors by Fast- ICA Algorithm)



(ICA basis vectors by ISA- gradient Algorithm)



(ICA basis vectors by T ICA Algorithm)



(ICA basis vectors by HO- ICA Algorithm)

## APPENDIX C

# Gray Scale Images

257 KB

35.2 KB

32 KB

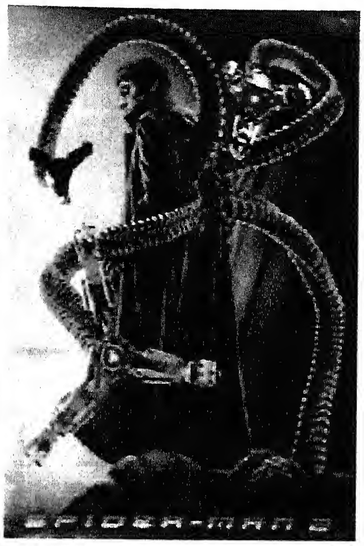
27.2 KB

7.79 KB

## Spider man2:

13.2 KB

11.5 KB



11.3 KB

## KK

49 KB

4.17 KB

# APPENDIX D

## Experimental Comparison

(1)

(2)

(3)

(4)

# References

1. M.E. Tipping and C.M. Bishop, "*Probabilistic principal component analysis*," Tech. Rep., Aston University, 1997.
2. J. H. Friedman. *Exploratory projection pursuit*. J. of the American Statistical Association, 82(397):249-266, 1987.
3. P.J. Huber. *Projection Pursuit. The Annals of Statistics*, 13(2): 435-475, 1985.
4. M.C. Jones and R. Sibson. *What is projection pursuit*? J. of the Royal Statistical Society, ser. A, 150:1-36, 1987.
5. M. Abramowitz and C.A. Stegun, Eds., *Handbook of Mathematical Functions and Formulas, Graphs and Mathematical Tables*, Dover, New York, 9 edition, 1972.
6. M. Girolami and C. Fyfe, "*Negentropy and kurtosis as projection pursuit indices provide generalized ICA algorithms*," in Advances in Neural Information Processing Systems, A. Cichocki and A. Back, Eds., 1996.
7. A. Hyvärinen, "*Fast ICA by a fixed-point algorithm that maximizes non-Gaussianity*," in Independent Component Analysis: Principles and Practice, S.J. Roberts and R. Everson, Eds., pp. 71–94. Cambridge University Press, 2001.
8. P. Comon. *Independent Component Analysis – a new concept?* Signal Processing, 36:287—314, 1994.
9. Hyvärinen and E.Oja. *A fast fixed-point algorithm for independent component analysis*. Neural Computation, 9(7):1483-1492, 1997.
10. E. Oja. *Subspace Methods of Pattern Recognition*. Research Studies Press, Letchworth, England, 1983.
11. Haykin, S. *Neural Networks, A Comprehensive Foundation*. Macmillan College Publishing Company, New York, 1994.
12. Papoulis. *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill, 3$^{rd}$ edition, 1991.
13. HYVÄRINEN, A. *A family of fixed-point algorithms for independent component analysis*. Tech. Rep. A40, Helsinki University of Technology, Faculty of Information Technology, Laboratory of Computer and Information Science, 1996.
14. HYVÄRINEN, A. *One unit contrast functions for independent component analysis*. A statistical Analysis, 1997 Submitted to a journal.

15. MENDEL, J. *Tutorial on higher-order statistics (spectra) in signal processing and system theory*: Theoretical results and some applications, Proceedings of the IEEE 79, 3(Mar. 1991), 278-305.

16. NIKIAS, C., and MENDEL, J. *Signal processing with higher-order spectra.* IEEE Signal Processing Magazine (July 1993), 10-37.

17. LUENBERGER, D. G. *Optimization by Vector Space Methods.* Series in decision and control. John Wiley and Sons, New York, 1969.

18. CAO, X,-R., AND Liu, R. –w. *General approach to blind source separation.* IEEE Transactions on Signal Processing 44, 3 (1996), 562—571.

19. T. M. Cover and J. A. Thomas. *Elements of Information Theory.* John Willey & Sons, 1991.

20. A. Cichocki, R.E. Bonger, L. Moszczynski, and K. Pope. *Modified Herault-Jutten algorithms for blind separation of sources.* Digital Signal Processing, 7:80-93, 1997.

21. A. Hyvärinen. *New approximations of differential entropy for independent component analysis and projection pursuit.* In Advances in Neural Information Processing Systems, volume 10, pages 273-279. MIT Press, 1998.

22. D.-T. Pham, P. Garrat, and C. Jutten. *Separation of a mixture of independent sources through a maximum likelihood approach.* In Proc. ÉUSIPCO, pages 771—774, 1992.

23. A.J. Bell and T.J. Sejnowski. *An information-maximization approach to blind separation and blind deconvolution.* Neural Computation, 7:1129-1159, 1995.

24. J.-P. Nadal and N. Parga. *Non-linear neurons in the low noise limit: a factorial code maximizes information transfer.* Network, 5:565-581, 1994.

25. J.-F. Cardoso. *Infomax and maximum likelihood for source separation.* IEEE Letters on Signal Processing, 4:112-114, 1997.

26. B. A. Perlmutter and L. C. Para. *Maximum likelihood blind source separation: A context-sensitive generalization of ICA.* In Advances in Neural Information Processing Systems, volume 9, pages 613-619, 1997.

27. T.-W. Lee, M. Girolami, and T. J. Sejnowski. *Independent component analysis using an extended infomax algorithm for mixed sub-Gaussian and super-Gaussian sources.* Neural Computation, 11(2): 417-441, 1999.

28. A. Hyvärinen and E. Oja. *Independent component analysis by general nonlinear Hebbian-like learning rules.* Signal Processing, 64(3):301-313, 1998.

29. J. H. Friedman and J.W. Turkey. *A projection pursuit algorithm for exploratory data analysis*. IEEE Trans. Of computers, c-23 (9):881-890, 1974.

30. C. Jutten and J. Herault. *Blind separation of sources, part I: An adaptive algorithm based on neuromimetic architecture*. Signal Processing, 24:1-10, 1991.

31. N. Delfosse and P. Loubaton. *Adaptive blind separation of independent sources: a deflation approach*. Signal Processing, 45:59—83, 1995.

32. A. Cichocki, R. E. Bonger, L. Moszczynski, and K. Pope. *Modified Herault-Jutten algorithms for blind separation of sources*. Digital Signal Processing, 7:80—93, 1997.

33. A. Cichocki and R. Unbehauen. *Robust neural networks with on-line learning for blind identification and blind separation of sources*. IEEE Trans. On Circuits and Systems, 43(11): 894—906, 1996.

34. A. Cichocki and R. Unbehauen, L. Moszczynski, and E. Rummert. *A new on-line adaptive algorithm for blind separation of source signals*. In Proc. Int. Symposium on Artificial Neural Networks ISANN-94, pages 406—411, Tainan, Taiwan, 1994.

35. Beate Laheld and Jean –François Cardoso. *Adaptive source separation with uniform performance*. In Proc. EUSIPCO, Pages 183—186, Edinburgh, 1994.

36. J.-F. Cardoso and B. Hvam Laheld. *Equivalent adaptive source separation*. IEEE Trans. On Signal Processing, 44(12):3017-3030, 1996.

37. E. Oja. *A simplified neuron model as a principal component analyzer*. J. of mathematical Biology, 15:267—273, 1982.

38. E. Oja. *Neural Networks, principal components, and subspaces*. Int. J. Neural Systems, 1:61—68, 1989.

39. E. Oja. And J. Karhunen. *On stochastic approximation of the eigenvectors and eigenvalues of the expectation of a random matrix*. Journal of Math. Analysis and Applications, 106: 69—84, 1985.

40. E. Oja. H. Ogawa, and J. Wangviwattana. *Learning in nonlinear constrained Hebbian networks*. In T. Kohonen et al. , editor, Artificial Neural Networks, Proc. ICANN'91, pages 385—390, Espoo, Finland, 1991. North-Holland, Amsterdam.

41. J. Karhunen, E. Oja, L. Wang, R. Vigário, and J. Joutsensalo. *A class of neural networks for independent component analysis*. IEEE Trans. On Neural Networks, 8(3):486-504, 1997.

42. E. Oja. *The nonlinear PCA learning rule in independent component analysis.* Neurocomputing, 17(1):25—46, 1997.

43. E. Oja. *Nonlinear PCA criterion and maximum likelihood in independent component analysis.* In Proc. Int. Workshop on Independent Component Analysis and Signal Separation (ICA'99), pages 143-148, Aussois, France, 1999.

44. L. – Y. Wang and Karhunen. *A unified neural bigradient algorithm for robust* PCA and MCA. Int. J. of Neural Systems, 7(1):53—67, 1996.

45. J. - F. Cardoso. *Eigen-structure of the fourth-order cumulant tensor with application to the blind source separation problem.* In Proc. ICASSP'90, pages 2655—2658, Albuquerque, NM, USA, 1990.

46. J. - F. Cardoso. *Super-symmetric decomposition of the fourth-order cumulant tensor. Blind identification of more sources than sensors.* In Proc. ICASSP'1991, pages 3109—3112, 1991.

47. J. - F. Cardoso. *Iterative techniques for blind source separation using only fourth-order cumulants.* In Proc. EUSIPCO, pages 739—742, Brussels, Belgium, 1992.

48. J. - F. Cardoso. and P. Comon. *Independent component analysis, a survey of some algebraic methods.* In Proc. ISCAS'96, volume 2, pages 93—96, 1996.

49. J. - F. Cardoso. and A. Souloumiac. *Blind beam forming for non Gaussian signals.* IEEE Proceedings- F, 140(6):362—370, 1993.

50. J. - F. Cardoso. *Source separation using higher order moments.* In Proc. ICASSP'89, pages 2109—2112, 1989.

51. A. Hyvärinen. *Fast and Robust fixed-point algorithms for independent component analysis.* IEEE Trans. On Neural Networks, 10(3): 626-634, 1999.

52. S.-I. Amari, A. Cichocki, and H. H. Yang. *A new learning algorithm for blind source separation.* In Advance in Neural Information Processing System 8, pages 757-763. MIT Press, Cambridge, MA, 1996.

53. A. D. Back and A.S. Weigend. *A first application of independent component analysis to extracting structure from stock returns.* Int. J. on Neural Systems, 8(4):473—484, 1998.

54. A. Hyvärinen. *The fixed-point algorithm and maximum likelihood estimation for independent component analysis.* Neural Processing Letters, 10(1): 1-5, 1999.

55. X. Giannakopoulos, J. Karhunen, and E. Oja. *Experimental comparison of neural ICA algorithms.* In Proc. Int. Conf. on Artificial Neural Network (ICANN'98), pages 651- 656, Skövde, Sweden, 1998.

56. R. Vigário, V. Jousmäki, M. Hämäläinen, R. Hari, and E. Oja. *Independent component analysis for identification of artifacts in magneto encephalographic recordings.* In Advances in Neural Information Processing Systems 10, pages 229—235. MIT Press, 1998.

57. T. Ristaniemi and J. Joutsensalo. *On the performance of blind source separation in CDMA downlink.* In Proc. Int. Workshop on Independent Component Analysis and Signal Separation (ICA'99), pages 437—441, Aussois, France, 1999.

58. Andreas Jung. *An introduction to a new data analysis tool: Independent Component Analysis*, Regensburg, March 18*th* 2002.

59. Wang, L., Karhunen, J., and Oja. E. *A bigradient optimization approach for robust PCA,MCA, and source separation*, In Proceedings of the IEEE International Conference on Neural Networks (ICNN)'95 (Perth,Australia,1996), pp. 1684-1689.

60. Chaur-Chin Chen, *On the Selection of Image Compression Algorithm*, National Tsing Hua University Hsinchu 300, Taiwan, 1998.

61. *Image Compression - from DCT to Wavelets*: A Review by Subhasis Saha.;